

**PROTOTIPE APLIKASI PENGOLAHAN SINYAL HRV
MENGUNAKAN MATLAB**

SKRIPSI

untuk memenuhi salah satu persyaratan
mencapai derajat Sarjana S1



Disusun oleh:

Badar Pradipta

13524070

**Jurusan Teknik Elektro
Fakultas Teknologi Industri
Universitas Islam Indonesia
Yogyakarta
2020**

LEMBAR PENGESAHAN

PROTOTIPE APLIKASI PENGOLAHAN SINYAL HRV MENGGUNAKAN MATLAB

TUGAS AKHIR

**Diajukan sebagai Salah Satu Syarat untuk Memperoleh
Gelar Sarjana Teknik
pada Program Studi Teknik Elektro
Fakultas Teknologi Industri
Universitas Islam Indonesia**

Disusun oleh:

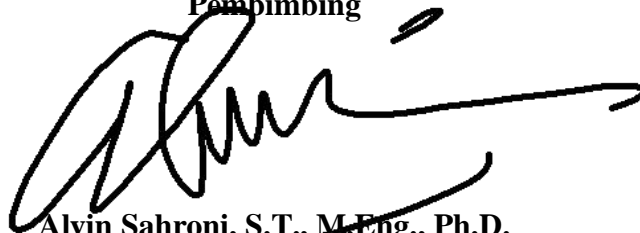
**Badar Pradipta
13524070**

الجامعة الإسلامية
ببغور

Bogor, 25-07-2020

Menyetujui,

Pembimbing



Alvin Sahroni, S.T., M.Eng., Ph.D.

NIK: 095240402

LEMBAR PENGESAHAN

SKRIPSI

PROTOTIPE APLIKASI PENGOLAHAN SINYAL HRV MENGGUNAKAN MATLAB

Dipersiapkan dan disusun oleh:

Badar Pradipta

13524070

Telah dipertahankan di depan dewan penguji

Pada tanggal: 13 Agustus 2020

Susunan dewan penguji

Ketua Penguji : Alvin Sahroni, S.T., M.Eng., Ph.D.

Anggota Penguji 1: Yusuf Aziz Amrullah, S.T., M.Sc., Ph.D.

Anggota Penguji 2: Firdaus, S.T., M.T.

Skripsi ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana

Tanggal: 24 Agustus 2020

Ketua Program Studi Teknik Elektro

Yusuf Aziz Amrullah, S.T., M.Sc., Ph.D.

045240101

PERNYATAAN

Dengan ini Saya menyatakan bahwa:

1. Skripsi ini tidak mengandung karya yang diajukan untuk memperoleh gelar kesarjanaan di suatu perguruan tinggi, dan sepanjang pengetahuan Saya juga tidak mengandung karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.
2. Informasi dan materi Skripsi yang terkait hak milik, hak intelektual, dan paten merupakan milik bersama antara tiga pihak yaitu penulis, dosen pembimbing, dan Universitas Islam Indonesia. Dalam hal penggunaan informasi dan materi Skripsi terkait paten maka akan diskusikan lebih lanjut untuk mendapatkan persetujuan dari ketiga pihak tersebut diatas.



Bogor, 25 Juli 2020



Badar Pradipta -

KATA PENGANTAR

Puji syukur penulis panjatkan kehadirat Allah Subhanahu wa Ta'ala yang telah memberikan segala nikmat, rahmat dan hidayah-Nya sehingga pada akhirnya penulis dapat menyelesaikan tugas akhir yang berjudul "PROTOTIPE APLIKASI PENGOLAHAN SINYAL HRV MENGGUNAKAN MATLAB". Tidak lupa shalawat dan salam senantiasa tercurahkan kepada junjungan kita Nabi Muhammad Shallallahu 'Alaihi wassallam yang telah menuntun kita dari zaman kegelapan ke zaman yang terang benderang ini. Tujuan penulisan skripsi ini untuk memenuhi sebagian syarat memperoleh gelar Sarjana Teknik (S.T.) bagi mahasiswa program S1 di program studi Teknik Elektro Fakultas Teknologi Industri Universitas Islam Indonesia.

Tugas akhir ini penulis selesaikan dengan mengikuti tahap-tahap pengerjaan dari teori yang telah penulis dapatkan selama masa studi. Selama proses penyelesaian tugas akhir ini penulis banyak memperoleh ilmu dan pengalaman yang sangat bermanfaat bagi penulis. Penulis menyadari sepenuhnya bahwa isi dari tugas akhir ini masih jauh dari sempurna karena keterbatasan ilmu dan pengetahuan penulis sebagai mahasiswa, oleh karena itu dengan segala kerendahan hati, penulis menerima saran dan kritikan yang sifatnya membangun dari semua pihak. Pada kesempatan ini penulis ingin menyampaikan rasa terima kasih kepada:

1. Allah Subhanahu wa Ta'ala dengan segala rahmat dan nikmat-Nya yang telah mengizinkan penulis menyelesaikan skripsi ini.
2. Kedua orang tua penulis, Ibu dan Bapak tercinta yang selalu mendukung, mendoakan, membiayai penulis dalam segala hal apapun.
3. Bapak Alvin Sahroni, S.T., M.Eng., Ph.D. selaku dosen pembimbing yang telah banyak sekali membantu, memberikan bimbingan dan pengarahan dengan sangat sabar kepada penulis dalam penyusunan tugas akhir ini.
4. Ketiga adik-adik yang selalu menjadi motivasi penulis.
5. Dosen-dosen dan karyawan Jurusan Teknik Elektro Fakultas Teknologi Industri UII, yang bersedia dengan sabar membantu dan berbagi ilmunya selama masa belajar dan pembuatan tugas akhir.
6. Teman-teman Teknik Elektro angkatan 2013 yang telah banyak membantu dan memberikan motivasi kepada penulis.

Akhir kata penulis sampaikan harapan semoga tugas akhir ini dapat memberikan manfaat khususnya bagi penulis dan pembaca pada umumnya. Semoga Allah Subhanahu wa Ta'ala selalu memberikan rahmat dan hidayah-Nya kepada kita semua. Aamiin.

Wassalaamu'alaikum warahmatullaahi wabarakaatuh

Bogor, 25 Juli 2020

Badar Pradipta



ARTI LAMBANG DAN SINGKATAN

BPM = *Beats Per Minute*

ECG = *Electrocardiogram*

PJK = *Penyakit Jantung Koroner*

PDI = *Program Dari Internet*

HRV = *Heart Rate Variability*

PTSD = *Posttraumatic Stress Disorder*

GUI = *Graphical User Interface*

SDNN/RR = *Standard Deviation Normal-to-Normal/R-to-R intervals*

CVRR = *The Coefficient of Variation of RR Intervals*

ULF = *Ultra Low Frequency*

VLF = *Very Low Frequency*

LF = *Low Frequency*

HF = *High Frequency*

NN50 = *Number of pairs of adjacent NN intervals, differing by more than 50 ms in the entire recording.*

pNN50 = *NN50 count divided by the total number of all NN intervals*

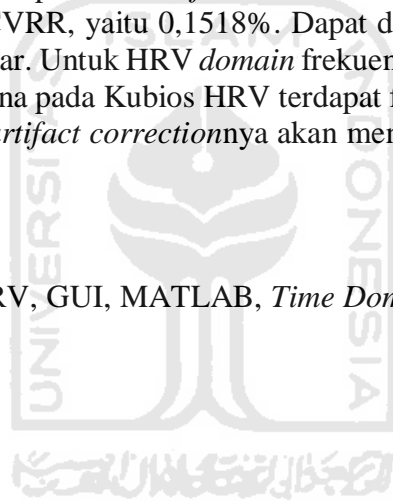
RMSSD = *The square Root of the Mean of the Sum of the Squares of Differences between adjacent NN intervals.*



ABSTRAK

Variabilitas detak jantung atau *heart rate variability* (HRV) adalah variasi dari interval antar detak jantung/RR Interval (interval antara puncak R-R dalam sinyal ECG). Secara umum analisis HRV bisa dilakukan dengan 2 metode, yaitu analisis HRV *domain* waktu (*time domain analysis*), dan *domain* frekuensi (*frequency domain analysis*). Di era digital ini, *raw data* ECG bisa ditemukan dengan mudah diberbagai halaman di internet, namun hanya sebatas *raw data* saja. Data yang disediakan tidak disertai dengan adanya perhitungan analisis nilai HRV. Berdasarkan temuan tersebut penulis terinspirasi membuat sebuah “Prototipe Aplikasi Pengolahan Sinyal HRV Menggunakan MATLAB”. Prototipe aplikasi ini menggunakan *Graphical User Interface* (GUI) MATLAB. *Format* data yang dapat dibaca dan diproses oleh aplikasi ini terbatas pada data dengan *format* .mat dan .txt. GUI dibuat dengan fitur berbagai *input from user*. GUI menampilkan nilai-nilai parameter HRV *domain* waktu antara lain Mean RR, SDRR, CVRR dan BPM serta parameter HRV *domain* frekuensi yaitu nilai *power* dari VLF, LF, dan HF. *Output* GUI dibandingkan dengan *tools* lain (*software* Kubios HRV, dan Program Dari Internet (PDI)). Dari 8 sample *raw data* ECG yang digunakan (4 data berdurasi 30 menit dan 4 data berdurasi 1 jam) untuk validasi, diketahui tidak ditemui perbedaan nilai BPM. Diketahui selisih perbandingan rata-rata nilai terbesar pada : (1) Mean RR, yaitu 0,1398 ms. (2) SDRR, yaitu 109,9175 ms. Selisih nilai yang cukup besar ini terjadi karena pada Kubios HRV terdapat fitur *Artifact correction*, yang akan mengoreksi puncak-puncak R yang tidak normal. (3) CVRR, yaitu 0,1518%. Dapat diambil kesimpulan perhitungan pada program GUI sudah relatif benar. Untuk HRV *domain* frekuensi terjadi perbedaan nilai *power* dan pola urutan pita frekuensi, karena pada Kubios HRV terdapat fitur *Artifact correction*. Hal ini menyebabkan setiap karakteristik *artifact correction*nya akan menghasikan nilai parameter HRV yang berbeda-beda.

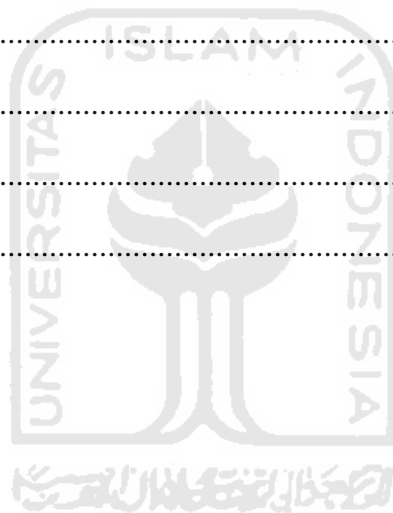
Kata Kunci : ECG, RR interval, HRV, GUI, MATLAB, *Time Domain*, *Frequency Domain*



DAFTAR ISI

LEMBAR PENGESAHAN	i
LEMBAR PENGESAHAN	ii
PERNYATAAN.....	iii
KATA PENGANTAR.....	iv
ARTI LAMBANG DAN SINGKATAN.....	vi
ABSTRAK.....	vii
DAFTAR ISI.....	viii
DAFTAR GAMBAR.....	x
DAFTAR TABEL.....	xii
BAB 1 PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah.....	2
1.4 Tujuan Penelitian.....	3
1.5 Manfaat Penelitian.....	3
BAB 2 TINJAUAN PUSTAKA	4
2.1 Studi Literatur	4
2.2 Tinjauan Teori.....	5
2.2.1 Detak Jantung.....	5
2.2.2 <i>Electrocardiogram</i> (ECG)	6
2.2.3 <i>Hear Rate Variability</i> (HRV) / Variabilitas Detak Jantung.....	7
2.2.4 Analisis HRV pada <i>domain</i> waktu (<i>time domain analysis</i>).....	7
2.2.5 Analisis parameter HRV pada <i>domain</i> frekuensi (<i>frequency domain analysis</i>)	9
BAB 3 METODOLOGI	13
3.1 Dataset yang digunakan	13

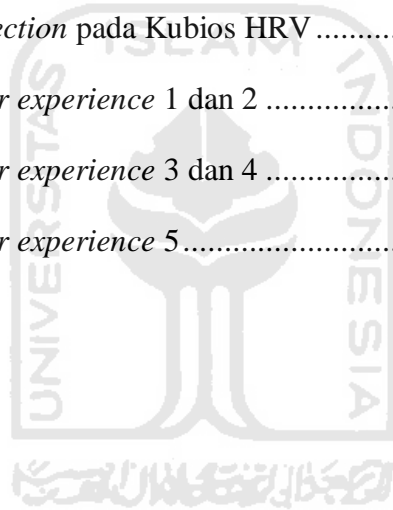
3.2 Perancangan sketsa <i>user interface</i>	13
3.3 Algoritme aplikasi dan penjelasannya	16
3.4 Metode validasi GUI.....	21
BAB 4 HASIL DAN PEMBAHASAN	22
4.1 Hasil Perancangan GUI	22
4.1.1 HRV <i>domain</i> waktu	25
4.1.2 HRV <i>domain</i> frekuensi	26
4.2 Validasi data.....	28
4.3 <i>User experience</i>	37
BAB 5 KESIMPULAN DAN SARAN	40
5.1 Kesimpulan	40
5.2 Saran	40
DAFTAR PUSTAKA.....	41
LAMPIRAN.....	44



DAFTAR GAMBAR

Gambar 2. 1 Morfologi Jantung[33].....	5
Gambar 2. 2 Morfologi sinyal EKG[34].....	6
Gambar 2. 3 Gelombang P, kompleks QRS, dan T [35].	6
Gambar 2. 4 Puncak-puncak R.....	9
Gambar 2. 5 <i>Plot RR time series 30 sample data (RR interval)</i>	10
Gambar 2. 6 Interpolasi Linear[36].....	10
Gambar 2. 7 RR interval yang melewati proses interpolasi lalu di <i>resampling</i> [37].	11
Gambar 2. 8 <i>Plot domain frekuensi (power spectral density) dengan pembagian 3 pita frekuensinya</i>	12
Gambar 3. 1 Sketsa awal <i>interface</i> menggunakan <i>paint</i>	13
Gambar 3. 2 Desain akhir <i>interface</i> GUI.....	14
Gambar 3. 3 Cara penggunaan GUI	14
Gambar 3. 4 <i>flow chart</i> algoritme sistem GUI.....	17
Gambar 3. 5 Cari data ECG	17
Gambar 3. 6 <i>Input frekuensi sampling</i>	17
Gambar 3. 7 Durasi Data ECG = Durasi Standar Analisis HRV ?	18
Gambar 3. 8 Data ECG sumbu-x detik/menit	18
Gambar 3. 9 <i>Input</i> durasi data ECG (detik) yang ingin di analisis HRVnya.....	19
Gambar 3. 10 <i>Input threshold</i>	19
Gambar 3. 11 RR interval	19
Gambar 3. 12 Data ECG > <i>threshold</i>	20
Gambar 3. 13 HRV <i>domain</i> waktu.....	20
Gambar 3. 14 HRV <i>domain</i> frekuensi.....	20

Gambar 4. 1 Hasil perancangan <i>interface</i> GUI.....	22
Gambar 4. 2 Hasil perancangan <i>interface</i> GUI - Fungsi perbesar plot PSD	23
Gambar 4. 3 Cara penggunaan GUI	23
Gambar 4. 4 HRV <i>domain</i> waktu.....	25
Gambar 4. 5 HRV <i>domain</i> frekuensi.....	26
Gambar 4. 6 <i>Output</i> dari GUI 1 (<i>file name</i> 103m.mat)	28
Gambar 4. 7 <i>Output</i> Kubios HRV dan PDI 1 (<i>file name</i> 103m.mat)	29
Gambar 4. 8 <i>Output</i> dari GUI 2 (<i>file name</i> 16273m.mat)	29
Gambar 4. 9 <i>Output</i> Kubios HRV dan PDI 2 (<i>file name</i> 16273m.mat)	30
Gambar 4. 10 <i>Artifact correction</i>	35
Gambar 4. 11 Contoh <i>artifact correction</i> pada Kubios HRV	35
Gambar 4. 12 Hasil pertanyaan <i>user experience</i> 1 dan 2	38
Gambar 4. 13 Hasil pertanyaan <i>user experience</i> 3 dan 4	38
Gambar 4. 14 Hasil pertanyaan <i>user experience</i> 5.....	39



DAFTAR TABEL

Tabel 3. 1 Komponen-komponen GUI.....	15
Tabel 4.1 Perbandingan hasil perhitungan parameter HRV	31



BAB 1

PENDAHULUAN

1.1 Latar Belakang Masalah

Variabilitas detak jantung atau *heart rate variability* (HRV) adalah variasi dari interval antar detak jantung (atau antara puncak R-R dalam data ECG) yang dihitung dalam satuan waktu [1]. HRV memberikan gambaran gejala fisiologis dari detak jantung, dan sebagai parameter yang berguna untuk memahami keseimbangan antara aktivitas 2 cabang sistem saraf otonom (saraf simpatis dan parasimpatis) [2]. Salah satu fungsi HRV yang sangat penting adalah untuk mengetahui tingkat stres dan kelelahan (pikiran maupun fisik) seseorang [3]. Untuk mengetahui karakteristik HRV seseorang, kita harus memiliki rekam jantungnya terlebih dahulu. Rekam jantung bisa dilakukan menggunakan alat yang bernama Elektrokardiograf, dari Elektrokardiograf ini akan dihasilkan sebuah sinyal Elektrokardiogram (ECG) yang dihasilkan oleh aktivitas listrik otot jantung. Dari sinyal ECG inilah seseorang dapat mengetahui informasi keadaan jantungnya apakah normal atau adanya suatu kelainan [4]. Pada umumnya, sinyal ECG ditampilkan melalui monitor atau dicetak pada kertas khusus ECG. Dari sinyal ECG tersebut, pengukuran yang paling sederhana untuk dilakukan adalah dengan menganalisis parameter *domain* waktu (*time domain analysis*). Dengan metode ini baik detak jantung pada titik waktu tertentu, ataupun interval antara 2 puncak R (dalam gelombang kompleks QRS) berturut-turut dapat ditentukan. Sedangkan metode *domain* frekuensi (*frequency domain analysis*) menggunakan analisis *power spectral density* (PSD) yang menyediakan informasi dasar tentang bagaimana daya (*power*) didistribusikan sebagai fungsi frekuensi [5].

Di berbagai halaman di internet, *raw data* ECG dapat dengan mudah ditemukan. Contohnya di halaman seperti Physionet dan Physiobank ATM dimana *raw data* ECG bisa dengan mudah di unduh dalam beberapa jenis *format file* seperti .mat, .dat, .csv. Namun, halaman internet tersebut tidak memberikan hasil perhitungan parameter HRV dari setiap data yang mereka sediakan.

Aplikasi analisis HRV yang sudah ada umumnya menggunakan metode *auto detection* sinyal ECG. Dari setiap *raw data* ECG yang di *inputkan*, nilai *threshold* dan durasi data ECG (yang ingin di analisis) sudah otomatis dihitung oleh aplikasi tersebut. Parameter (*threshold*) untuk mendeteksi puncak sinyal ECG tidak dapat diubah. Pada umumnya, aplikasi seperti ini juga terhubung dengan perangkat keras yang digunakan. Hal ini menyebabkan aplikasi tidak dapat digunakan oleh orang lain dengan konfigurasi yang berbeda pula.

Berdasarkan temuan tersebut, penulis terinspirasi untuk membuat *Graphical User Interface* (GUI) untuk menganalisis HRV dari *raw data* ECG menggunakan MATLAB. Penulis menggunakan MATLAB karena dasar-dasar pemrogramannya penulis dan rekan-rekan lainnya pelajari selama duduk di bangku perkuliahan. Dan MATLAB adalah bahasa pemrograman *universal* untuk para *engineer* di seluruh dunia. GUI ini khusus untuk mengolah *raw data* ECG dengan jenis *file* .mat dan .txt saja. Penulis membuat GUI ini bertujuan agar *user* dapat mengetahui parameter-parameter HRV dari *raw data* ECG yang mereka miliki. Target pengguna adalah orang awam (bukan dari bidang elektro/medis) yang memiliki data rekam jantung dalam bentuk data digital (*file* .mat / .txt). *Interface* GUI dibuat semenarik dan menyenangkan mungkin, agar pada *user* mudah dalam menggunakannya.

Kontribusi perancangan GUI ini juga diharapkan dapat membantu menganalisis HRV dari *device-device* lainnya yang menyimpan *raw data* ECG dalam bentuk *format file* yang sesuai dengan GUI ini (.mat/.txt). Sehingga untuk pengembangan kedepannya GUI bisa dijadikan aplikasi *universal* untuk menganalisis HRV.

1.2 Rumusan Masalah

1. Bagaimana merancang sebuah *Graphical User Interface* (GUI) untuk mengetahui parameter HRV dalam *domain* waktu (BPM, Mean RR, SDRR, dan CVRR) dan *domain* frekuensi (*power* VLF, LF, dan HF) menggunakan MATLAB ?
2. Bagaimana validasi perancangan GUI yang penulis buat terhadap aplikasi yang telah ada (tersebar) saat ini yang disediakan secara gratis?

1.3 Batasan Masalah

1. Penelitian ini hanya untuk mengetahui parameter HRV dalam *domain* waktu (BPM, Mean RR, SDRR, dan CVRR) dan *domain* frekuensi (*power* VLF, LF, dan HF).
2. *Graphical user interface* (GUI) ini hanya untuk data ECG dengan format *file* .mat dan .txt.
3. *Graphical user interface* (GUI) ini menggunakan software MATLAB.
4. Minimal (standar) durasi data ECG yang bisa digunakan untuk analisis HRV adalah 5 menit [1][6].

1.4 Tujuan Penelitian

Merancang sebuah *graphical user interface* (GUI) untuk mengetahui parameter HRV dalam *domain* waktu (BPM, Mean RR, SDRR, dan CVRR) dan *domain* frekuensi (*power* VLF, LF, dan HF) menggunakan MATLAB.

1.5 Manfaat Penelitian

1. Untuk membantu dalam penelitian-penelitian tentang analisis HRV di masa yang akan datang.
2. Sebagai prototipe aplikasi analisis HRV yang cocok untuk berbagai macam *device* akuisisi data rekam denyut jantung.



BAB 2

TINJAUAN PUSTAKA

2.1 Studi Literatur

Penelitian yang dilakukan oleh Theodorus L Hartono, Dalu Setiaji, Iwan Setyawan bertujuan untuk menggabungkan *software* dan *hardware* untuk merekam sinyal jantung sekaligus menganalisis HRV dari rekam jantung tersebut. Mereka menggunakan program Kubios HRV *Analysis Software 2.0* dan GUI MATLAB. Mereka menampilkan nilai parameter HRV dalam *domain* waktu dan *domain* frekuensi dengan GUI MATLAB. Untuk *domain* waktu parameter HRV yang mereka tampilkan nilainya adalah NN, mean NN, SDNN, rMSSD, NN50, dan pNN50. *Raw data* ECG yang mereka gunakan diambil dari internet [7].

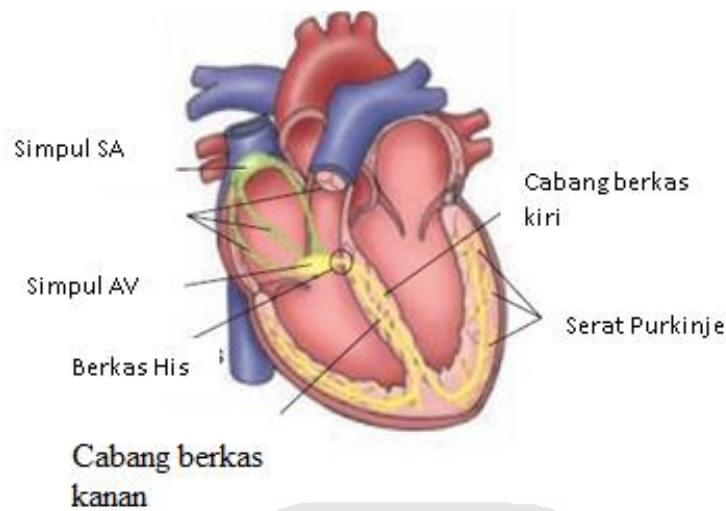
GUI MATLAB yang mereka buat cukup detail dalam menjabarkan nilai parameter HRV, tetapi durasi waktu data ECG yang di analisis sudah ditentukan oleh program GUI. *User* tidak dapat memilih sendiri durasi waktu dari *raw data* ECG yang ingin mereka ketahui parameternya. Mereka juga tidak menampilkan sebuah nilai dasar yang biasa digunakan sebagai parameter untuk menentukan keadaan jantung seseorang, yaitu nilai detak jantung per menit (BPM).

Penelitian lain dilakukan Agung, Ratna, dan Farhan dengan cara mendeteksi puncak R dari beberapa sampel data ECG. Sampel data ECG yang mereka gunakan adalah data dari MIT-BIH *Arrhythmia Database*, data ini bisa didapat dari halaman Physionet secara gratis. Ada beberapa tahapan yang harus mereka lakukan sebelum dapat mengklasifikasikan hasil dari data ECG yang digunakan, yaitu pemfilteran sinyal, deteksi sinyal kompleks QRS, penurunan sinyal, penguadratan sinyal, deteksi puncak R, deteksi RR interval, perhitungan BPM, dan yang terakhir adalah proses klasifikasi seluruh hasilnya.

Hasil akhir dari penelitian mereka adalah, klasifikasi antara sampel data ECG yang memiliki jumlah denyut jantung normal atau tidak normal/aritmia (bradikardia atau nilai BPM kurang dari 60 dan takikardia atau nilai BPM lebih dari 100). Mereka menghitung RR interval tetapi tidak menghitung parameter HRV seperti NN/RR, mean NN/RR, SDNN/SDRR, rMSSD, NN50, dan pNN50. Mereka menggunakan fitur RR interval hanya untuk menentukan nilai BPM dari tiap sampel data yang mereka gunakan. Mereka juga hanya menggunakan pemrograman MATLAB biasa tanpa bantuan *software* lain [8]. Program mereka akan sulit jika ingin digunakan oleh orang lain, karena mereka tidak menggunakan GUI.

2.2 Tinjauan Teori

2.2.1 Detak Jantung



Gambar 2. 1 Morfologi Jantung[33].

Gambar 2.1 adalah gambar morfologi jantung. Berikut adalah tahapan proses jantung berdetak. Jantung kita berkontraksi disebabkan oleh rangsangan dari simpul sinoatrial (SA) yang terletak pada bagian atas serambi kanan. Simpul atrioventikular (AV) terletak diantara serambi kanan dan bilik kanan. Berfungsi untuk menghantarkan impuls (listrik) dari serambi ke bilik. Impuls dari simpul AV diteruskan ke seluruh bilik melalui berkas HIS (diberi nama sesuai nama penemunya, Wilhelm His Jr). Berkas HIS adalah kumpulan sel otot jantung yang dikhususkan untuk konduksi listrik. Sebagai bagian dari sistem konduksi listrik jantung, berkas HIS berfungsi untuk mentransmisikan impuls listrik dari AV node[9][10]. Di ujung berkas HIS terdapat banyak cabang-cabang yang disebut serat *Purkinje*, yang bertugas untuk meneruskan impuls dari HIS ke seluruh otot bilik jantung. Hasilnya, bilik akan berkontraksi sehingga darah akan dipompa keluar dari bilik dan mengalir ke sistem peredaran darah[11]. Ketika jantung melakukan proses memompa darah keseluruh tubuh, satu kali siklus proses itu disebut satu detak jantung.

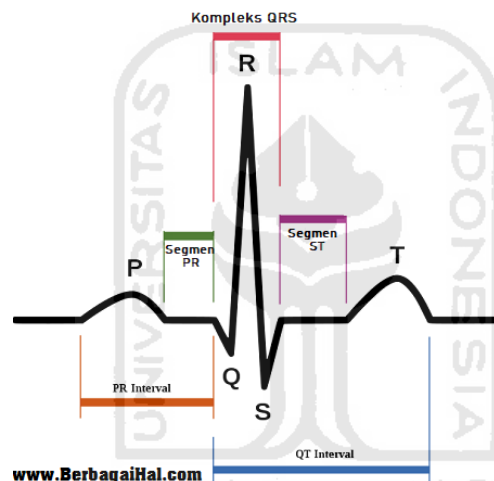
Detak jantung atau denyut jantung adalah seberapa banyak jantung kita berdenyut/berdetak dalam satu menitnya (*beats per minute*) [12]. Detak jantung seseorang dapat menginformasikan kondisi kesehatan fisiknya sendiri. Banyaknya detak jantung setiap orang berbeda-beda tergantung suhu udara, posisi tubuh, emosi, ukuran tubuh, sedang menggunakan obat atau tidak, itu semua dapat mempengaruhi denyut jantung [13]. Detak jantung dengan irama normal biasa disebut normal sinus *rhythm* dalam keadaan istirahat/santai adalah 60-100 *beats per minute* (BPM). Sedangkan detak jantung dengan irama yang tidak normal disebut aritmia. Aritmia terbagi menjadi

dua yaitu takikardia (detak jantung berdetak lebih cepat dari detak jantung normal) dan brikardia (detak jantung berdetak lebih lambat dari detak jantung normal) [12].

2.2.2 Electrocardiogram (ECG)

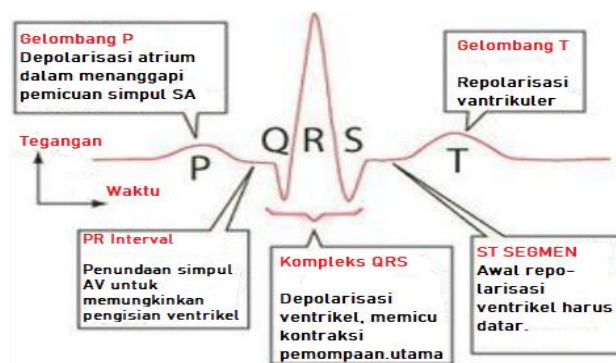
Electrocardiogram (ECG) atau dalam bahasa Indonesianya Elektrokardiogram (EKG) adalah suatu sinyal yang dihasilkan oleh aktivitas listrik pada jantung, yang diukur oleh sebuah alat yang bernama Elektrokardiograf [14]. Sedangkan elektrokardiografi adalah ilmu yang mempelajari hasil rekaman dari aktivitas listrik jantung pada permukaan tubuh [15].

ECG bekerja dengan cara mengukur potensial listrik sebagai fungsi waktu yang dihasilkan oleh jantung. Potensial listrik dihasilkan oleh pemicu denyut jantung, yang dapat merubah sistem kelistrikan jantung. Beda potensial inilah yang divisualisasikan sebagai sinyal/grafik ECG, dan dicetak langsung pada kertas khusus ECG atau hanya ditampilkan pada monitor [16].



Gambar 2. 2 Morfologi sinyal EKG[34].

Gelombang P menggambarkan depolarisasi atrial yang dipicu oleh node SA, segmen PR menunjukkan berhentinya impuls pada AV node, gelombang QRS menggambarkan depolarisasi ventrikel, segment ST menggambarkan tidak adanya impuls yang disebabkan adanya periode refrakter di sel miokardium, dan gelombang T menggambarkan repolarisasi pada aktivitas gerak jantung [17].



Gambar 2. 3 Gelombang P, kompleks QRS, dan T [35].

Untuk menuliskan sinyal ECG harus menggunakan kertas khusus tersendiri yang digunakan untuk menggambar hasil aktivitas listrik pada jantung. Kertas ECG dengan ukuran 25 mm/s memiliki 2 kotak, yaitu kotak besar dan kotak kecil. Dimana 1 kotak besar sama dengan 5 kotak kecil. 1 kotak besar memiliki ukuran 0,2 sec (bidang horizontal) x 5 mm (bidang vertikal). Dan 1 kotak kecil memiliki ukuran 0,04 sec (bidang horizontal) x 1 mm (bidang vertikal) [18].

Dengan pesatnya digitalisasi data, data ECG juga sudah banyak sekali yang tersedia dalam *format* digital seperti dalam *format file* .mat, .txt, .dat, .csv dll. Halaman internet Physionet adalah salah satu penyedia *raw data* ECG terbesar dan terlengkap. Disana terdapat data ECG dengan berbagai keadaan yang berbeda-beda pada setiap individunya. Ada detak jantung normal, ada yang aritmia, dll. Keunggulan data ECG yang disimpan dalam bentuk digital adalah dapat dengan mudah untuk diproses dan analisis menggunakan berbagai komputer dengan berbagai macam aplikasi gratis yang tersedia banyak di internet.

2.2.3 Heart Rate Variability (HRV) / Variabilitas Detak Jantung

Heart rate variability (HRV) atau variabilitas detak jantung adalah variasi dari interval antara detak jantung (atau antara gelombang R-R dalam ECG) berturut-turut yang dihitung dalam satuan waktu [1]. Sinyal ECG standar terdiri dari 3 gelombang yaitu gelombang P, kompleks QRS, dan gelombang T. Dari ketiga gelombang ini puncak R memiliki amplitudo paling besar, dan hal inilah yang menyebabkan mengapa pengukuran HRV dihitung dari nilai interval antara puncak R dengan puncak R (RR interval) [6]. Sedangkan interval dari gelombang kompleks QRS yang berdekatan biasa disebut dengan interval N-N (normal-normal) [19].

HRV berguna untuk memahami keseimbangan antara aktivitas 2 cabang sistem saraf otonom (saraf simpatis dan parasimpatis) dan bisa merepresentasikan keadaan fisik dan mental seseorang [2]. Ketika tubuh sedang melakukan aktifitas fisik berat seperti berlari, maka sistem saraf simpatis akan mengambil alih dengan membuat jantung berdetak lebih cepat, dan sebaliknya ketika tubuh dalam keadaan santai, giliran sistem saraf parasimpatis yang mengambil alih, yang akan membuat jantung menjadi lebih pelan dalam berdetak [20]. Saat ini, sudah banyak ditemukan alat-alat yang dapat mengukur HRV dengan instan. Pengukuran HRV secara umum sendiri dapat dilakukan dengan dua metode, yaitu metode *domain* waktu dan metode *domain* frekuensi [5].

2.2.4 Analisis HRV pada *domain* waktu (*time domain analysis*)

Analisis HRV pada *domain* waktu mengukur interval antara detak jantung (dalam siklus jantung normal) secara berturut-turut. Biasanya dari perekaman ECG selama 24 jam setiap sinyal

kompleks QRS akan terdeteksi, yang berarti RR interval juga terdeteksi. Dari nilai RR interval inilah parameter-parameter HRV dari suatu perekaman data ECG dapat diketahui, seperti Mean RR, SDRR, BPM, dan CVRR[21]. Ke empat nilai parameter HRV *domain* waktu tersebut adalah nilai yang akan ditampilkan pada GUI ini, berikut adalah penjelasannya :

1. Mean RR interval

Mean RR interval adalah nilai rata-rata dari seluruh RR interval (dalam rentang durasi tertentu). Mean RR adalah salah satu nilai yang akan penulis tampilkan pada GUI, karena berhubungan dengan perhitungan parameter BPM. Berikut persamaannya.

$$\bar{x} = \frac{x_1 + x_2 + x_3 \dots x_n}{n} \quad (2.1)$$

2. *Beats per minute* (BPM)

BPM adalah parameter yang utama dan harus diketahui sebelum melakukan analisis HRV maupun karakteristik denyut jantung manusia. Persamaan untuk mencari parameter BPM adalah sebagai berikut [8][6].

$$\text{BPM} = \frac{60}{\bar{x}} \quad (2.2)$$

3. *Standard Deviation Normal-to-Normal/R-to-R intervals* (SDNN / SDNN)

SDNN/SDRR Adalah standar deviasi dari seluruh NN interval / RR interval. Berikut persamaannya

$$\text{SDRR} = \sqrt{\frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n - 1}} \quad (2.3)$$

4. *The Coefficient of Variation of RR Intervals* (CVRR)

CVRR adalah rasio/perbandingan antara SDRR dengan Mean RR, nilai parameter ini dalam persentase (%). Berikut persamaannya [22][23].

$$\text{CVRR} = \frac{\text{SDRR}}{\text{Mean RR}} \% \quad (2.4)$$

Keterangan :

\bar{x} = Mean RR interval

$x_1 + x_2 + x_3 \dots x_n$ = data RR interval ke 1,2,3 sampai data ke n.

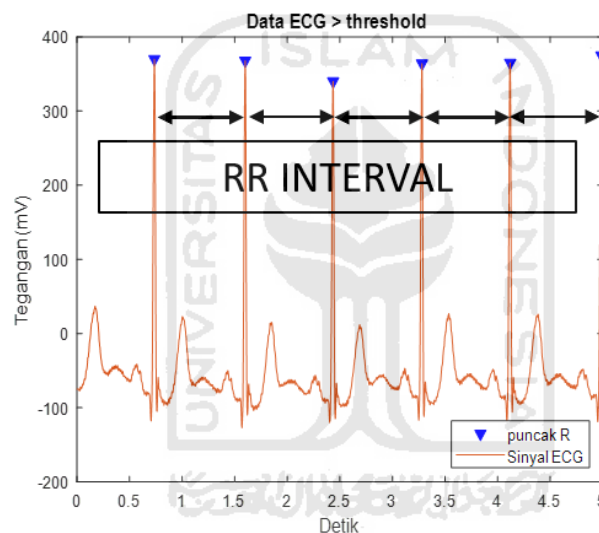
n = jumlah RR interval

x_i = data RR interval ke i

i = mewakili bilangan data ke 1,2,3 dan seterusnya

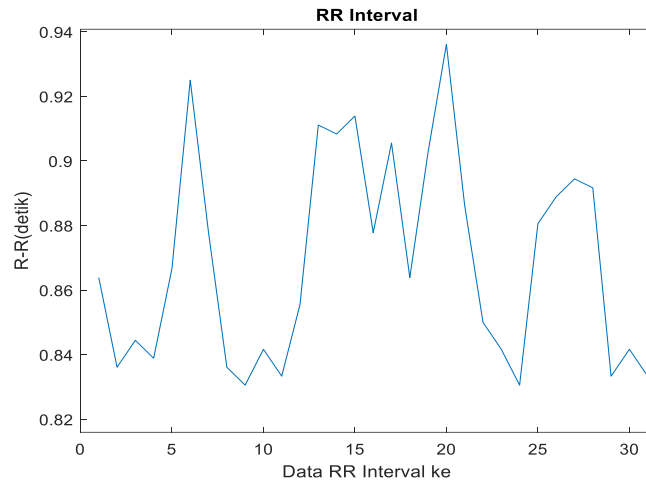
2.2.5 Analisis parameter HRV pada domain frekuensi (*frequency domain analysis*)

Setelah puncak-puncak R dapat dideteksi. Langkah selanjutnya adalah mencari interval antara puncak R ke puncak R terdekat (RR interval). Gambar 2.4 adalah contoh dari puncak-puncak R yang sudah terdeteksi.



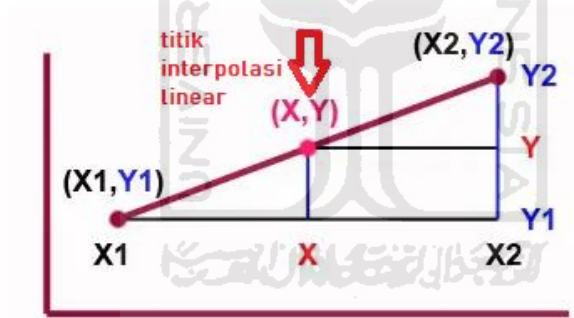
Gambar 2. 4 Puncak-puncak R

Bisa dilihat pada gambar 2.4 ada jarak antar puncak R satu dengan puncak R yang lain. Jarak dari setiap puncak R ke puncak R lain (dalam durasi waktu tertentu) akan disimpan dalam sebuah variabel data oleh MATLAB. Yang jika variabel itu di *plot* akan menjadi data RR *time series* (gambar 2.5 hanya menampilkan 30 *sample data RR time series*) seperti pada gambar 2.5.



Gambar 2. 5 Plot RR time series 30 sample data (RR interval)

Data RR time series ini belum dapat digunakan untuk menganalisis domain frekuensi HRV. Masih dibutuhkan beberapa proses lagi. Selanjutnya data RR time series harus di *resampling*. Sebelum di *resampling* RR time series harus melewati tahapan interpolasi linear terlebih dahulu. Interpolasi linear secara umum adalah cara mencari nilai di antara 2 data berdasarkan persamaan linear (garis lurus). Contoh kasus interpolasi linear seperti pada gambar 2.6.



Gambar 2. 6 Interpolasi Linear[36].

Dari gambar 2.6 didapatkan sebuah nilai persamaan seperti berikut

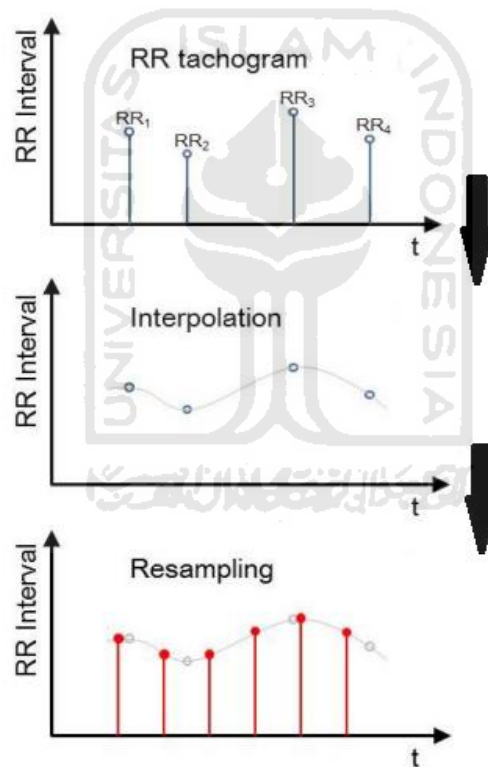
$$\frac{(X - X1)}{(X2 - X1)} = \frac{(Y - Y1)}{(Y2 - Y1)} \quad (2.5)$$

Dari persamaan tersebut nilai X dan Y bisa ditentukan dengan persamaan berikut

$$Y = Y1 + \frac{(X - X1)}{(X2 - X1)}(Y2 - Y1) \quad (2.6)$$

$$X = X1 + \frac{(Y - Y1)}{(Y2 - Y1)}(X2 - X1) \quad (2.7)$$

X dan Y adalah titik yang dicari menggunakan teknik interpolasi linear. Interpolasi linear diterapkan pada data RR *time series* untuk menurunkan/meningkatkan resolusi (jumlah) data pada titik-titik tertentu. Selanjutnya data yang sudah di interpolasi akan di *resampling*. Karena *range* pada sumbu-x (frekuensi) pada plot PSD nantinya hanya dibutuhkan 0 – 0,5 Hz. Maka data RR *time series* akan di *resampling* senilai 2 kali nilai maksimal sumbu-x yaitu, 0,5 Hz x 2 = 1 Hz [24]. RR *time series* akan di *resampling* menjadi 1 Hz (walaupun pada plot PSD yang ditampilkan GUI hanya 0 – 0,5 Hz saja). Data RR interval yang melalui proses interpolasi linear dan *resampling* secara garis besar bisa dilihat pada gambar 2.7 .



Gambar 2. 7 RR interval yang melewati proses interpolasi lalu di *resampling* [37].

Selanjutnya nilai tersebut diproses kembali menggunakan algoritme FFT. Terakhir adalah membagi frekuensi yang sudah didapatkan melalui serangkaian proses tadi menjadi 3 pita frekuensi.

Dari pembagian pita frekuensi inilah fungsi dari sistem saraf simpatis dan parasimpatis dapat diketahui. Disepakati bahwa saraf simpatis dan saraf parasimpatis memiliki rentang frekuensi tertentu yakni sebagai berikut:

1. Neuro-hormonal, berada pada *Very Low Frequency*, <0.04 Hz.
2. Syaraf Simpatis, berada pada *Low Frequency*, 0.04-0.15 Hz.
3. Syaraf Parasimpatis, berada pada *High Frequency*, 0.15-0.4 Hz [25].

Nilai power pada setiap pitanya (VLF, LF, dan HF) adalah nilai penjumlahan pada sumbu Y (di *range* masing-masing pita). Nilai *power* dihitung dengan persamaan sebagai berikut :

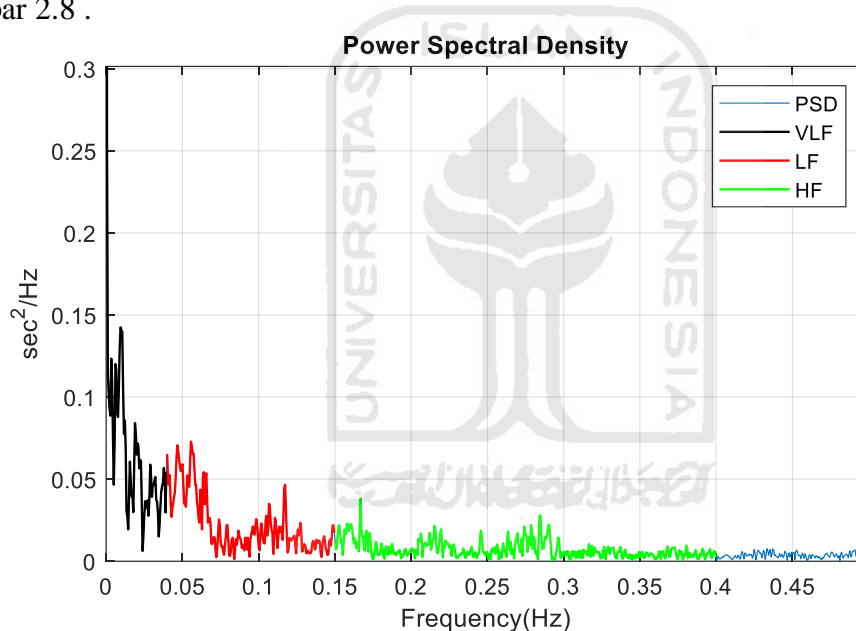
$$\text{Power (di setiap frekuensinya)} = P_1 + P_2 + P_3 + \dots P_n \quad (2.8)$$

Keterangan :

$P_1 + P_2 + P_3 + \dots P_n$ = nilai data *power* ke 1, 2, 3 sampai n pada *domain* frekuensi.

n = jumlah data *power* pada *domain* frekuensi.

Setelah RR interval melewati proses interpolasi linear dan *resampling*. Lalu hasilnya di proses kembali menggunakan algoritma FFT dan dibagi menjadi 3 pita frekuensi dan di *plot* seperti pada gambar 2.8 .



Gambar 2. 8 *Plot* domain frekuensi (*power spectral density*) dengan pembagian 3 pita frekuensinya

BAB 3

METODOLOGI

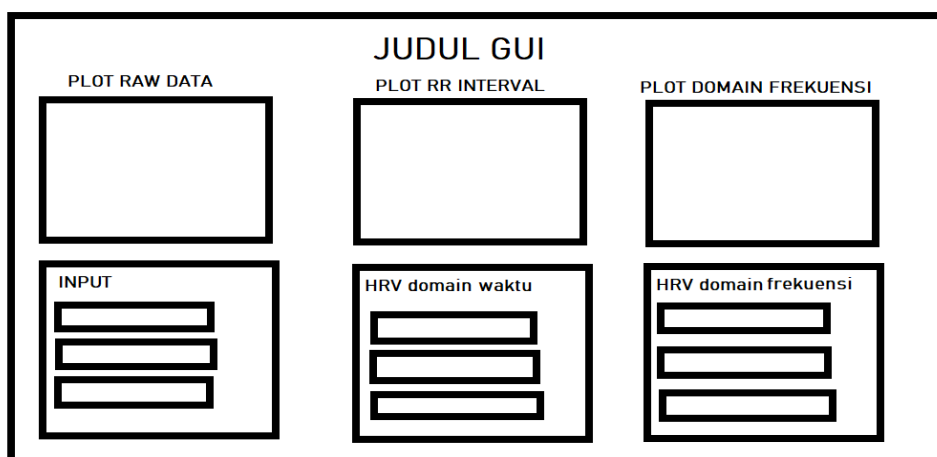
3.1 Dataset yang digunakan

Raw data ECG yang digunakan adalah yang sinyalnya teridentifikasi dengan jelas pada bagian sinyal P, kompleks QRS, dan sinyal T (khususnya puncak R). Ada 8 sampel *raw data* ECG yang digunakan, 4 sampel dengan durasi 60 menit (tidak ada detail keterangan dari *website* Physionet) dan 4 sampel dengan durasi 30 menit (sampel 1 adalah laki-laki dan sampel 2-4 adalah perempuan, semua sampel dalam keadaan sedang mengonsumsi obat). Semua *raw data* ECG yang digunakan untuk validasi diambil dari *website* <https://archive.physionet.org/cgi-bin/atm/ATM>. *Raw data* yang diambil detailnya sebagai berikut :

1. MIT-BIH *Arrhythmia database* (mitdb) dengan durasi 30 menit (panjang data 650000 kolom x 2 baris), dan dengan frekuensi *sampling* 360 (Hz).
2. MIT-BIH *Normal Sinus Rhythm Database* (nsrdb) dengan durasi 60 menit (panjang data 460800 kolom x 2 baris), dan dengan frekuensi *sampling* 128 (Hz).

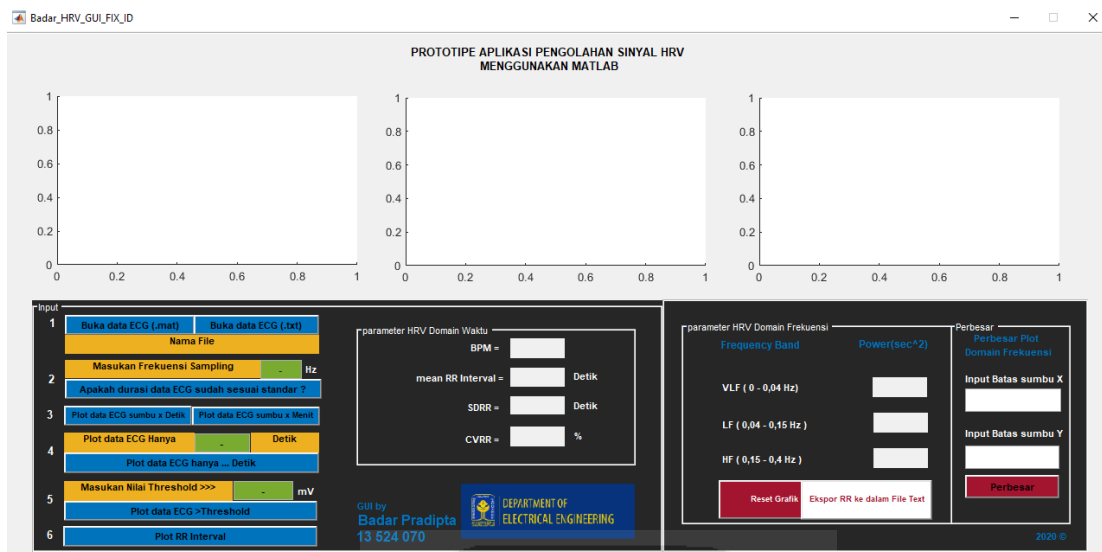
3.2 Perancangan sketsa *user interface*

Sebelum membuat GUI penulis harus mendesain terlebih dahulu sketsa *interfacenya*, karena *interface* yang bagus dan menarik menjadi daya tarik sendiri bagi *user*. Kenyamanan *user* dalam menggunakan aplikasi juga harus diperhatikan, agar *user* tidak cepat bosan dan betah berlama-lama dalam menggunakan aplikasi ini. Bisa dilihat pada gambar 3.1 adalah sketsa awal *interface*.



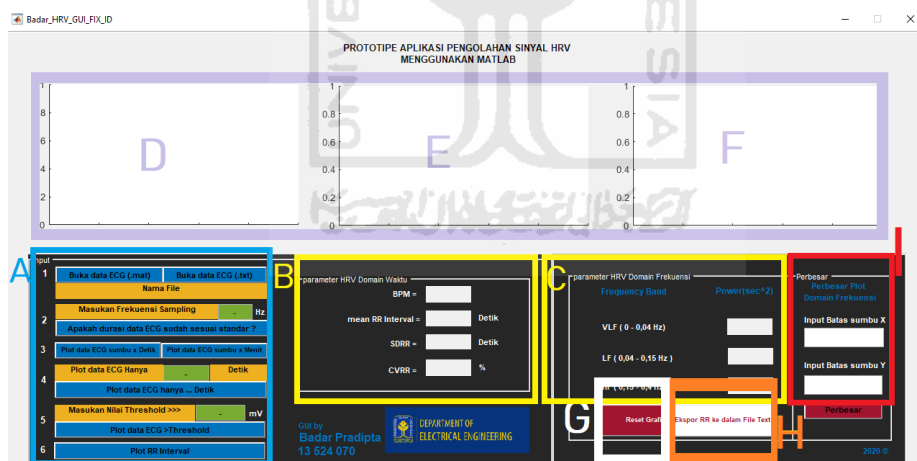
Gambar 3. 1 Sketsa awal *interface* menggunakan *paint*

Dari realisasi sketsa awal GUI, penulis mendapatkan banyak ide-ide untuk membuat GUI menjadi lebih menarik dan *simple* namun tetap informatif. Setelah melakukan beberapa kali perubahan, akhirnya desain akhir GUI sudah berhasil dirancang seperti pada gambar 3.2.



Gambar 3. 2 Desain akhir *interface* GUI

GUI dibuat semudah mungkin untuk digunakan oleh para *user*. Dapat dilihat pada gambar 3.3 adalah realisasi dari rancangan dan cara penggunaan GUI. Penjelasan tentang cara kerja dari rancangan GUI dilambangkan dengan huruf kapital, yang detailnya adalah sebagai berikut



Gambar 3. 3 Cara penggunaan GUI

- Bagian A adalah komponen-komponen yang menampung semua nilai *inputan* pada GUI. Mulai dari nilai *input* frekuensi *sampling* (f_s), *input* durasi *raw data ECG* yang ingin di analisis (detik), dan *input* nilai *threshold*. Nilai *threshold* adalah nilai ambang batas, atau suatu nilai yang dijadikan batasan minimal (amplitudo) dari sebuah data ECG yang dianggap sebagai puncak R..
- Bagian B adalah *output* dari GUI, yaitu nilai-nilai parameter HRV pada *domain* waktu (BPM, Mean RR, SDRR, dan CVRR).

- Bagian C adalah *output* dari GUI, yaitu nilai parameter HRV pada *domain* frekuensi (*power* VLF, LF, dan HF).
- Bagian D adalah komponen (*axes*) yang akan menampilkan hasil *plotting* dari *raw data* ECG, data ECG dalam sumbu-x waktu (detik & menit) dan data ECG yang nilainya lebih besar dari nilai *threshold*.
- Bagian E adalah komponen (*axes*) yang menampilkan grafik hasil *plotting* nilai RR interval.
- Bagian F adalah komponen (*axes*) yang menampilkan grafik hasil *plotting* dari *output* GUI, yaitu hasil komponen spektrum frekuensi (HRV *domain* frekuensi) dari RR interval (*time series* - pada bagian E) dalam satuan sec^2 / Hz (*power spectral density*).
- Bagian G adalah komponen yang berfungsi untuk *reset* seluruh grafik pada GUI.
- Bagian H adalah komponen yang berfungsi untuk melakukan ekspor nilai RR interval yang telah berhasil di ekstraksi oleh GUI ke dalam bentuk *file text*.
- Bagian I adalah komponen yang berfungsi untuk memperbesar gambar *plotting* spektrum frekuensi (*power spectral density*) pada bagian F.

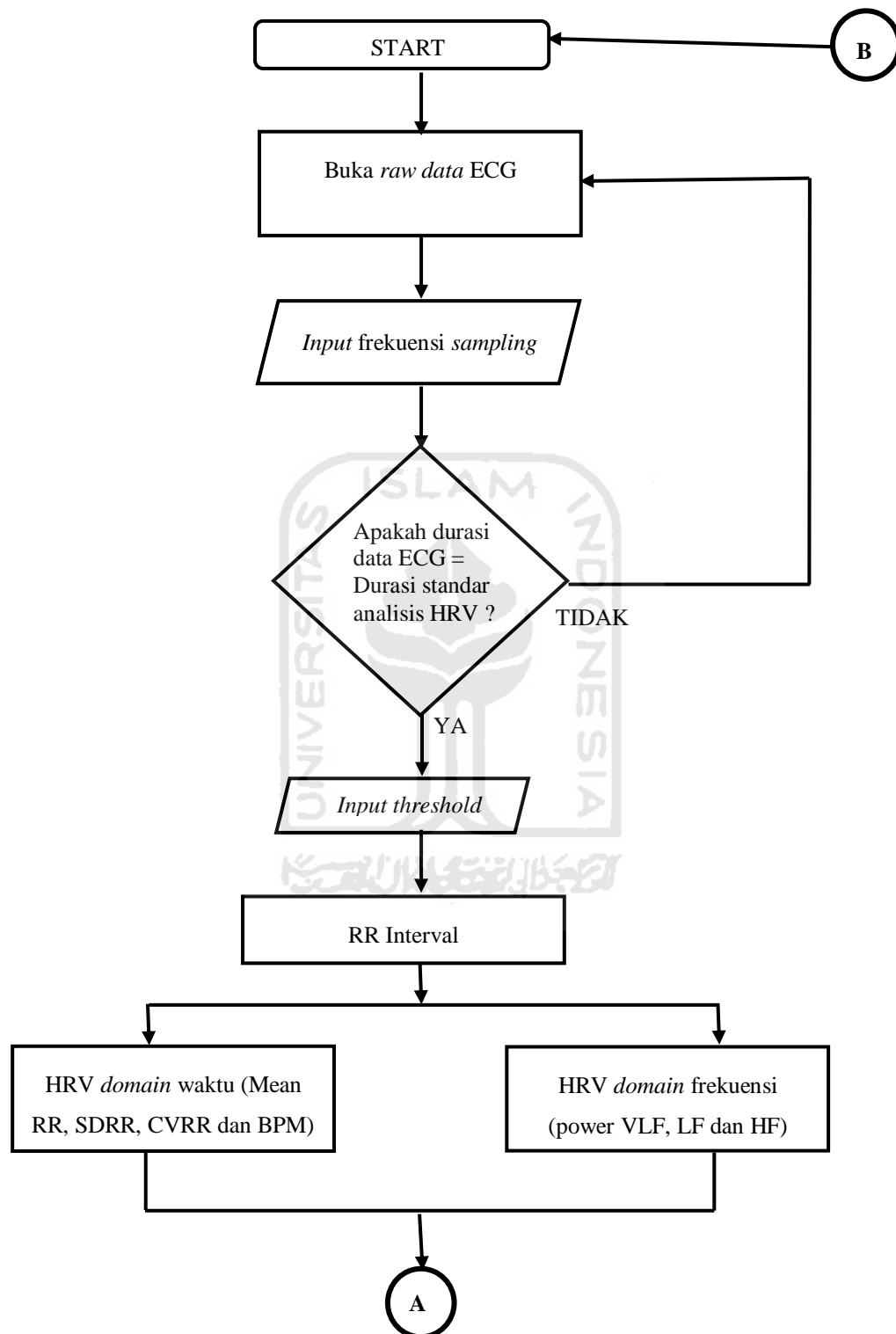
GUI dibuat dari kumpulan komponen-komponen dan perintah program yang mempermudah *user* dalam menggunakan sebuah program. Berikut adalah komponen-komponen yang digunakan dalam pembuatan GUI ini.

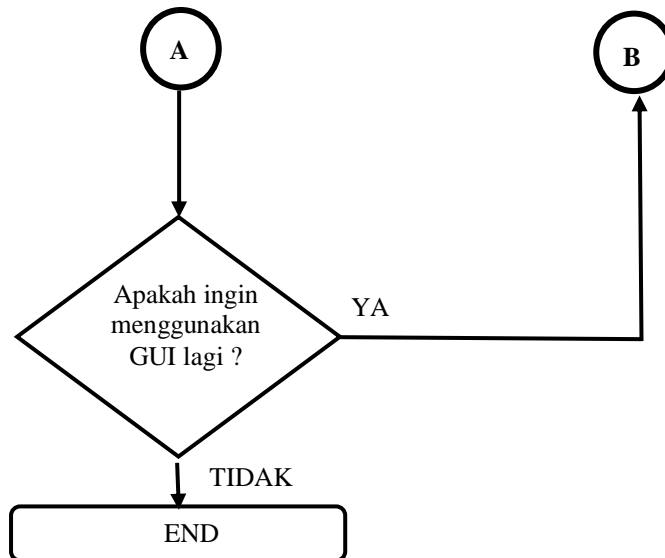
Tabel 3. 1 Komponen-komponen GUI

Nama Komponen	Jumlah	Fungsi
<i>Push Button</i>	11	Komponen ini berfungsi sebagai tombol eksekusi seluruh fitur-fitur yang ada pada GUI jika komponen ini di <i>klik</i> . Komponen ini terletak pada gambar 3.3 diatas pada bagian A, G, H, dan I.
<i>Axes</i>	4	Komponen ini berfungsi untuk menampilkan grafik hasil <i>plotting</i> dari program ini sesuai parameter yang dimasukan pada GUI. Komponen ini terletak pada gambar 3.3 diatas pada bagian D, E, dan F.
<i>Edit Text</i>	12	Komponen ini berfungsi untuk menampung nilai variabel <i>input</i> pada GUI. Ini terdapat pada gambar 3.3 diatas pada bagian A dan I.
<i>Static Text</i>	34	Komponen ini berfungsi untuk menampilkan nilai <i>output</i> dari perhitungan pada program GUI. Komponen ini terdapat pada gambar 3.3 diatas pada bagian A, B dan C.

3.3 Algoritme aplikasi dan penjelasannya

Berikut adalah *flow chart* algoritme bagaimana sistem GUI dapat bekerja.





Gambar 3. 4 *flow chart* algoritme sistem GUI

Berikut adalah penjelasan dari *flow chart* algoritme sistem GUI.

1. Cari Data ECG

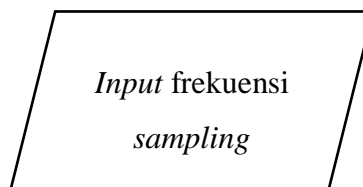


Gambar 3. 5 Cari data ECG

Tahapan pertama adalah mencari/membuka *raw data* ECG yang ingin di analisis. Setelah *raw data* ECG dipilih, GUI akan memprosesnya sebagai *input*. Lalu GUI akan *plot* data tersebut secara otomatis. GUI hanya bisa memproses *file* .mat dan .txt saja. *Raw data* ECG yang digunakan harus memiliki durasi standar minimal analisis HRV yaitu 5 menit [1][6].

2. *Input* frekuensi *sampling*

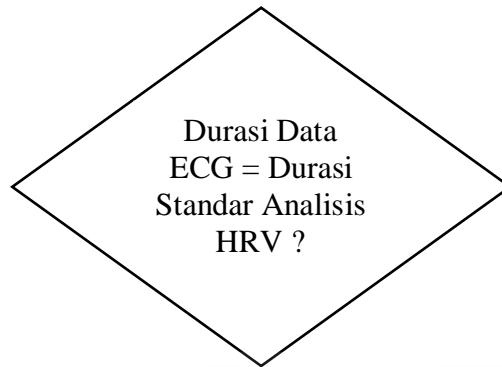
Selanjutnya *user* harus menginput frekuensi *sampling* (fs) dari data ECG yang digunakan. *Input* fs berfungsi untuk menjadikan sumbu-x (pada *plot* grafik) menjadi satuan waktu (detik/menit).



Gambar 3. 6 *Input* frekuensi *sampling*

3. Durasi Data ECG = Durasi Standar Analisis HRV ?

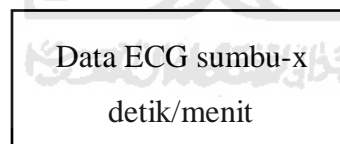
Setelah nilai *fs* di *input*. Selanjutnya GUI akan memproses dan mengecek data ECG yang digunakan. Jika durasi data ECG yang akan digunakan sudah sesuai durasi standar untuk analisis HRV (5 menit) akan muncul *warning box* yang menginformasikan bahwa data ECG dapat digunakan. Dan jika data ECG belum sesuai akan muncul *warning box* yang menginformasikan bahwa data ECG belum dapat digunakan. *User* harus mencari data ECG baru.



Gambar 3. 7 Durasi Data ECG = Durasi Standar Analisis HRV ?

4. Plot Data ECG sumbu-x detik dan menit

Setelah ada pemberitahuan data ECG dapat digunakan, sumbu-x data ECG akan dirubah menjadi detik atau menit. Tahapan ini juga bisa digunakan untuk mengecek durasi data ECG dalam sumbu-x detik apakah sudah sesuai dengan sumbu-x menit atau belum. Misalnya durasi data ECG 5 menit, berarti jika di *plot* dalam sumbu-x detik harus 300 detik dan dalam sumbu-x menit harus 5 menit. Jika kurang/lebih berarti ada perhitungan yang salah pada program.



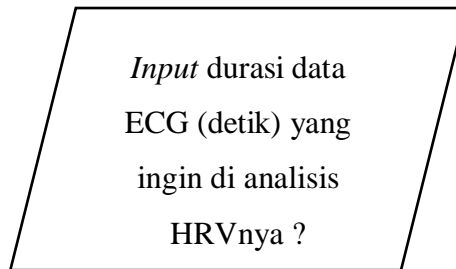
Gambar 3. 8 Data ECG sumbu-x detik/menit

5. *Input* durasi data ECG yang ingin di analisis

Pada tahapan ini *user* dapat memilih durasi (awal) data ECG yang ingin di analisis. Contoh, *user* hanya ingin menganalisis 5 menit awal dari total 1 jam durasi data ECG. *User* harus menginput durasi 5 menit ini ke dalam GUI. Dengan syarat, 5 menit tersebut dirubah ke dalam detik terlebih dahulu ($5 \times 60 = 300$ detik).

Tahapan ini juga berfungsi untuk memplot data ECG hanya beberapa detik saja, misalnya 3 detik saja. Ini berfungsi agar *user* mudah dalam menentukan nilai *threshold*. Karena jika hanya di *plot* beberapa detik saja, bentuk *raw data* ECG pasti terlihat sangat jelas dibandingkan dengan menentukan *threshold* dari data ECG yang berdurasi 30 menit atau 1 jam. Pasti akan sangat sulit. Hal ini dikarenakan sampel per sampel dari data ECG akan sangat berdekatan dan sangat rapat.

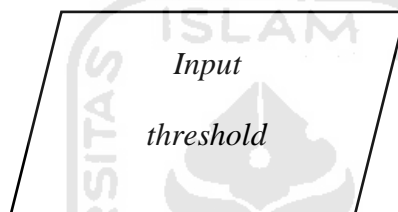
Nilai *threshold* tidak selalu harus nilai rata-rata dari sebuah kelompok data. Tidak ada parameter khusus dalam menentukan nilai *threshold* pada GUI. Dari fakta inilah kemungkinan besar hasil HRV dari setiap *user* akan berbeda.



Gambar 3. 9 *Input durasi data ECG (detik) yang ingin di analisis HRVnya*

6. *Input threshold*

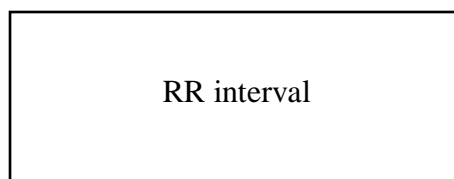
Selanjutnya *inputkan* pada GUI nilai *threshold* yang tadi sudah ditentukan pada tahapan sebelumnya.



Gambar 3. 10 *Input threshold*

7. Nilai RR interval

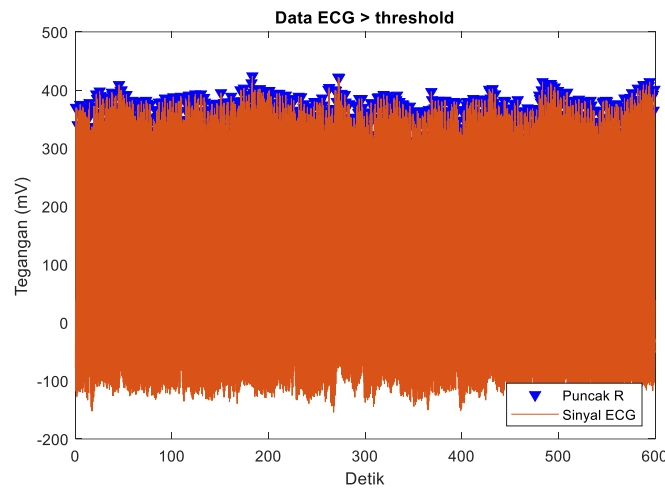
Tahapan ini dapat disebut tahapan utama pada GUI. Karena setelah melewati beberapa tahapan, *raw data* ECG berubah menjadi data nilai RR interval. Nilai RR interval juga akan di plot secara otomatis oleh GUI pada tahapan ini. Nilai RR interval bisa di ekspor kedalam *file text* (untuk digunakan pada penelitian lain) dengan mengklik tombol “Ekspor RR ke dalam *File Text*”



Gambar 3. 11 RR interval

RR interval adalah nilai utama dari GUI. Dari nilai RR interval, nilai HRV *domain* waktu dan *domain* frekuensi dapat diketahui. Dari *raw data* ECG yang di *input*, ditentukanlah sebuah nilai *threshold*. *Raw data* ECG yang sama dengan atau melebihi nilai *threshold* di asumsikan sebagai puncak R. Bisa dilihat pada gambar 3.12 adalah plotting data $ECG > threshold$. Puncak R disini di asumsikan sebagai sebuah detak jantung. Setiap jantung berdetak akan menghasilkan 3 sinyal yaitu, sinyal P, sinyal kompleks QRS, dan sinyal T. Dari ketiganya yang memiliki amplitudo

terbesar adalah puncak R. Itulah mengapa puncak R bisa di asumsikan sebagai sebuah detak jantung. Contoh hasil plotting data RR interval bisa dilihat pada gambar 2.5.



Gambar 3. 12 Data ECG $>$ *threshold*

Gambar 3.12 adalah hasil *plotting* data ECG yang lebih besar dari nilai *threshold* dengan durasi 300 detik/5 menit. Untuk lebih jelas data dari sampel ke sampelnya, penulis memperbesar *plot* tersebut pada gambar 2.4. Penulis memperbesar *plot*nya pada durasi 5 detik, agar antara puncak R dan puncak R lainnya terlihat jelas.

8. HRV *domain* waktu

GUI akan mengekstrak kembali nilai RR interval dengan algoritmenya dan memprosesnya menjadi nilai HRV *domain* waktu seperti Mean RR, SDRR, CVRR, dan BPM akan ditampilkan pada GUI.

HRV *domain* waktu (Mean RR,
SDRR, CVRR, dan BPM)

Gambar 3. 13 HRV *domain* waktu

9. HRV *domain* frekuensi

GUI juga akan mengekstrak kembali nilai RR interval dengan algoritmenya dan memprosesnya menjadi nilai HRV *domain* frekuensi seperti *power* VLF, *power* LF, dan *power* HF.

HRV *domain* frekuensi (*power*
VLF, *power* LF, dan *power* HF.)

Gambar 3. 14 HRV *domain* frekuensi

3.4 Metode validasi GUI

GUI akan dibandingkan dengan beberapa *free analysis software* HRV, yaitu Kubios HRV dan PDI (hanya menghitung HRV *domain* waktu). Nilai yang dibandingkan adalah nilai HRV *domain* waktu yaitu Mean RR, SDRR, CVRR dan BPM dan nilai HRV *domain* frekuensi seperti *power* VLF, LF, dan HF. *Input* pada Kubios HRV dan PDI adalah data RR Interval (*output* dari GUI). Sedangkan *input* pada GUI adalah *raw data* ECG yang diambil dari *website* physionet.

Sebelum dibandingkan antara ketiganya, satuan dari setiap parameter HRV *domain* waktu dan *domain* frekuensi harus disamakan terlebih dahulu. Parameter Mean RR, SDRR dan *power* pada setiap frekuensi (VLF, LF dan HF) pada Kubios HRV dan PDI semuanya menggunakan satuan *millisecond* (ms), sedangkan pada GUI adalah *second* (sec). Maka satuan GUI harus dirubah terlebih dahulu menjadi *millisecond* (ms), agar validasi dapat dilakukan.

Perbandingan nilai dilakukan dengan cara mencari selisih nilai diantara ketiganya. Dapat dihitung dengan cara, hasil GUI dikurangi hasil Kubios HRV dan hasil PDI. Selisih nilai yang didapat adalah nilai perbandingan perbedaan antara ketiganya.

Untuk HRV *domain* waktu pada GUI akan dibandingkan/validasi dengan perincian sebagai berikut :

1. BPM dibandingkan dengan Kubios HRV dan PDI (dihitung manual).
2. Mean RR dibandingkan dengan Kubios HRV dan PDI
3. SDRR dibandingkan dengan Kubios HRV dan PDI.
4. CVRR karena tidak ada pembandingnya, penulis akan membuat pembandingannya sendiri dengan membuat perbandingan manual antara ketiganya (GUI, Kubios HRV, dan PDI).

Untuk HRV *domain* frekuensi pada GUI hanya akan dibandingkan dengan Kubios HRV. Pada pembagian 3 pita frekuensi yang dilihat adalah pola urutan frekuensinya, *power* pada frekuensi mana yang paling besar, sedang, dan kecil. Lalu akan diambil kesimpulan apakah pola urutan *power* frekuensi dari yang terbesar ke yang terkecil pada GUI sama dengan pola urutan *power* frekuensi Kubios HRV atau tidak.

Untuk mengetahui seberapa besar respon pengguna, maka dilakukan evaluasi *user experience* dengan memberikan *form online* dengan beberapa parameter yang dijadikan sebagai penilaiannya seperti kenyamanan, kemudahan, dan kepuasan *user* saat menggunakan GUI.

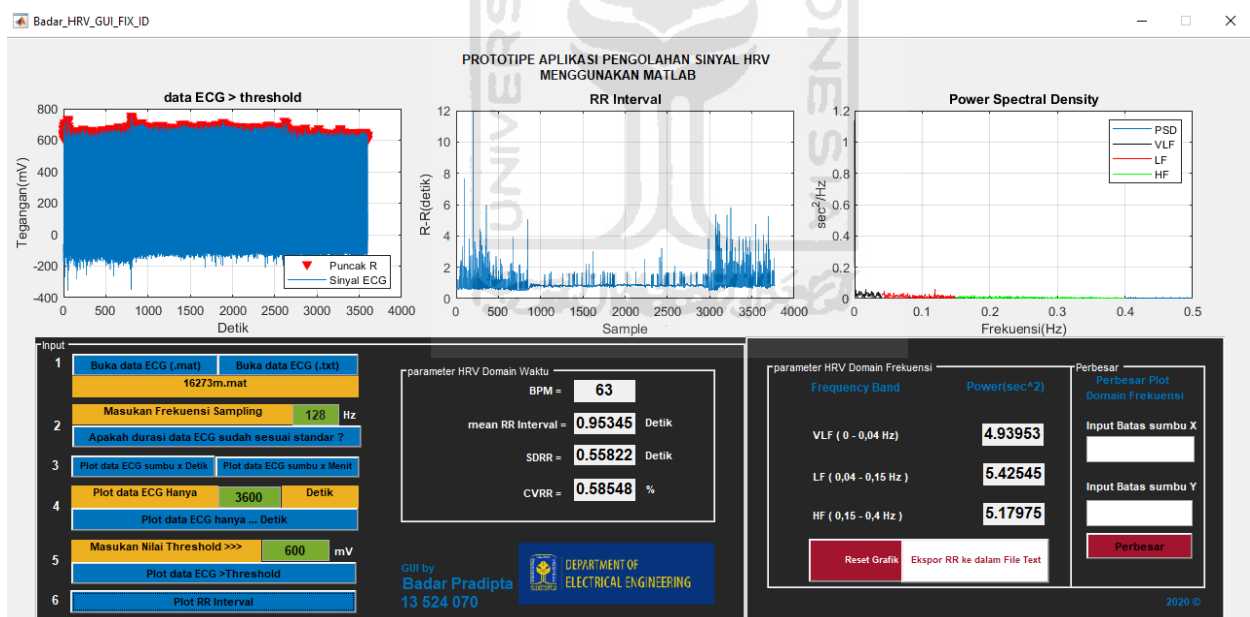
BAB 4

HASIL DAN PEMBAHASAN

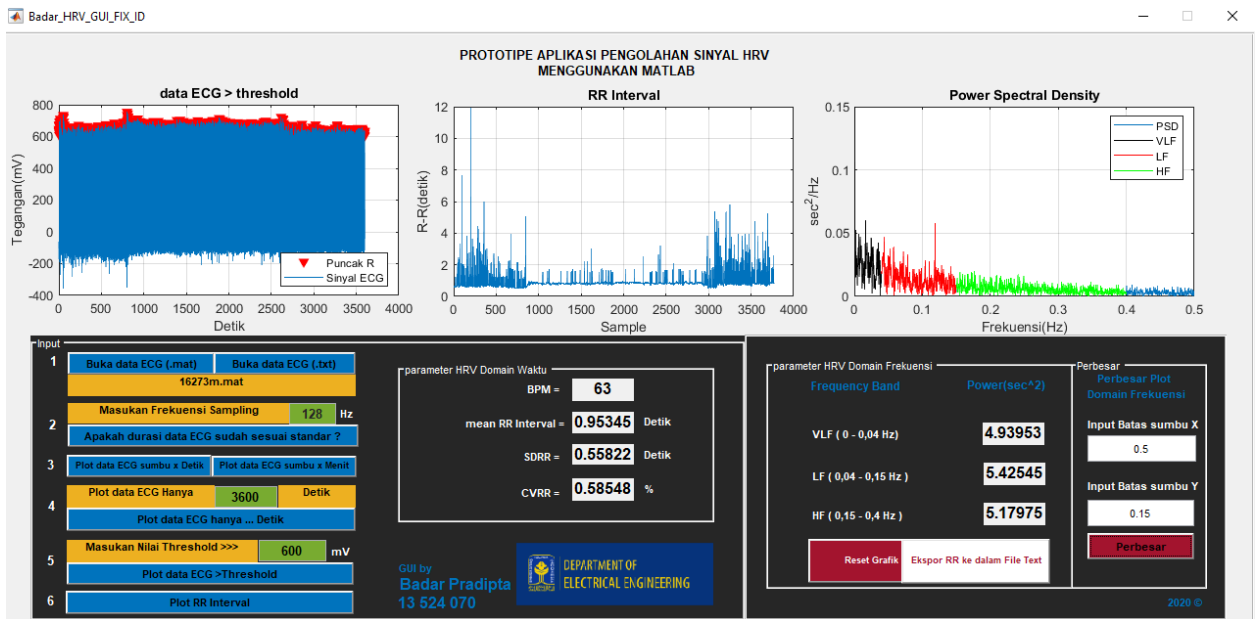
4.1 Hasil Perancangan GUI

Setelah mengalami beberapa kali perubahan desain, akhirnya desain akhir *interface* GUI sudah ditetapkan seperti pada gambar 4.1. GUI hanya bisa memproses *file* ECG dengan *format file* .mat dan .txt. Selain menampilkan nilai HRV *domain* waktu (BPM, Mean RR, SDRR, & CVRR) dan HRV *domain* frekuensi (VLF, LF, & HF), GUI juga menampilkan plot dari HRV *domain* frekuensi (*power spectral density*) dan plot dari RR interval.

Bisa dilihat pada gambar 4.1, GUI menampilkan nilai HRV *domain* waktu, seperti BPM, Mean RR, SDRR, dan CVRR beserta *plot* RR intervalnya, dan juga nilai HRV *domain* frekuensi, seperti *power* dari VLF, LF, & HF beserta *plot power spectral density* (PSD). Sayangnya *plot* PSD kurang terlihat jelas. Dari fakta inilah penulis menambahkan fungsi perbesar pada GUI, yang berguna untuk memperbesar *plot* PSD seperti pada gambar 4.2. *User* bebas untuk memilih berapa batas *plot* untuk sumbu x dan sumbu y.



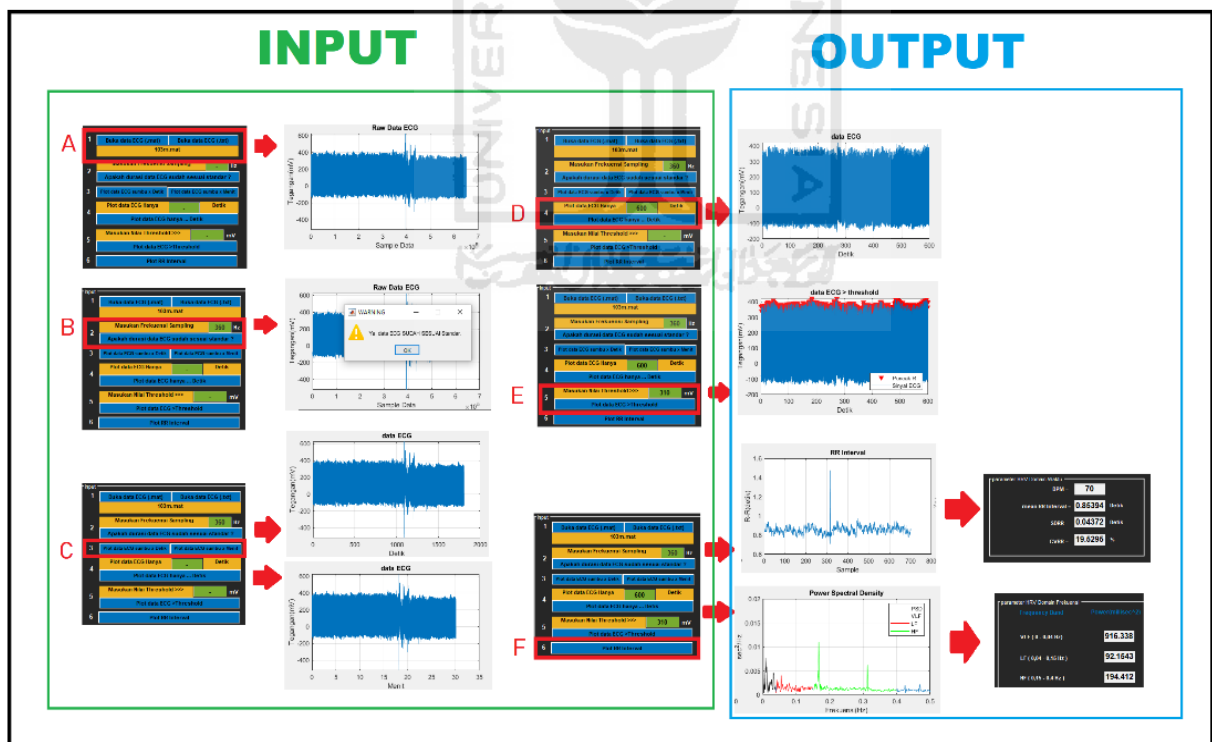
Gambar 4. 1 Hasil perancangan *interface* GUI



Gambar 4. 2 Hasil perancangan *interface* GUI - Fungsi perbesar plot PSD

Simulasi GUI dan cara pembacaan *outputnya*

Simulasi penggunaan GUI secara garis besar bisa dilihat pada gambar 4.3. Berikut adalah penjelasannya.



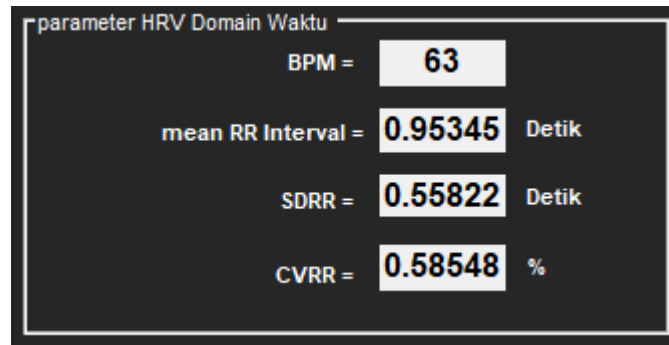
Gambar 4. 3 Cara penggunaan GUI

- Gambar A, adalah langkah pertama pada penggunaan GUI. *User* harus membuka data ECG yang mereka punya (*file .mat/.txt*). Hasil *plottingnya* terlihat pada gambar disebelahnya (yang ditunjukkan oleh tanda panah merah ke kanan).

- Gambar B, *user* harus menginput nilai frekuensi *sampling*, agar sumbu-x pada plot grafik berubah dari *sample data* menjadi satuan waktu (detik). Frekuensi *sampling* juga berfungsi untuk memastikan apakah durasi data ECG sudah sesuai dengan durasi standar minimal analisis HRV yaitu 5 menit atau tidak (karena 5 menit adalah *short term* dalam analisis HRV, sedangkan *long term* adalah 24 jam) [1][6]. *Warning box* akan muncul untuk memberitahu apakah data ECG sudah sesuai standar atau belum. Jika sudah sesuai standar data dapat digunakan, dan jika tidak sesuai maka data ECG tidak dapat digunakan.
- Gambar C, *user* dapat memplot data ECG dalam sumbu-x detik atau menit. Fungsinya untuk meyakinkan bahwa durasi data ECG yang digunakan sudah sesuai dengan durasi standar data ECG untuk analisis HRV (5 menit). Hasil *plottingnya* terlihat pada gambar disebelahnya (yang ditunjukkan oleh tanda panah merah ke kanan).
- Gambar D, selanjutnya adalah memilih durasi data ECG yang ingin di analisis. Misalnya durasi total data ECG adalah 30 menit, tetapi *user* hanya ingin mengetahui parameter HRV untuk durasi 5 menit saja, maka *user* harus merubah satuan menit ke dalam satuan detik, lalu nilai satuan detik di *inputkan* ke dalam GUI. Atau bisa juga digunakan untuk memplot hanya beberapa detik saja, misalnya 5 detik. Ini berfungsi untuk meyakinkan *user* ketika menentukan nilai *threshold*. Karena semakin kecil durasi maka sampel per sample data ECG pasti terlihat lebih jelas. Berbeda jika data ECG di plot dengan durasi langsung 30 menit atau 1 jam, pasti akan sulit untuk menentukan nilai *threshold* dari sampel data yang berdekatan dan rapat. Hasil *plottingnya* terlihat pada gambar disebelahnya (yang ditunjukkan oleh tanda panah merah ke kanan).
- Gambar E, *inputkan* nilai *threshold* yang tadi sudah ditentukan. Hasil *plottingnya* terlihat pada gambar disebelahnya (yang ditunjukkan oleh tanda panah merah ke kanan).
- Gambar F, atau tahapan terakhir, yaitu plot RR interval. Ketika *push button plot RR interval* di *klik*, maka nilai HRV *domain* waktu (BPM, Mean RR, SDRR, dan CVRR) serta HRV *domain* frekuensi (*power* VLF, LF, dan HF) akan otomatis ditampilkan oleh GUI. *Axes* juga akan otomatis memplot nilai RR interval dan HRV *domain* frekuensi (*power spectral density*).

GUI memiliki dua nilai *output*, yaitu HRV *domain* waktu dan HRV *domain* frekuensi, cara pembacaan outputnya adalah sebagai berikut.

4.1.1 HRV *domain waktu*



Gambar 4. 4 HRV *domain waktu*

1. BPM

BPM bisa merepresentasikan keadaan tubuh seseorang sedang sehat/tidak sehat. Nilai normal BPM manusia adalah 60 – 100 BPM. Jika kurang atau lebih dari nilai normal, seseorang dapat dikatakan memiliki irama detak jantung tidak normal (aritmia). Aritmia sendiri terbagi menjadi 2, yaitu bradikardia (detak jantung dibawah 60 BPM). Dan Takikardia (detak jantung diatas 100 BPM). Bradikardia biasanya terjadi pada orang-orang yang sudah berumur lanjut usia, yang kondisi fisiknya sudah menurun, atau sedang dalam keadaan mengkonsumsi obat-obatan tertentu. Sedangkan takikardia biasanya terjadi pada seseorang yang banyak melakukan aktifitas fisik secara terus menerus (olahraga) [12]. Yang akan membuat jantung bekerja dengan keras, dan mengakibatkan jantung berdetak lebih banyak dari biasanya [26].

2. Mean RR

Mean RR atau rata-rata nilai RR interval, adalah nilai rata-rata dari RR interval (dalam rentang durasi tertentu). RR interval sendiri adalah jarak/interval antara suatu puncak R (sinyal ECG) ke puncak R lainnya. Atau bisa juga di asumsikan jarak antara suatu detak jantung ke detak jantung berikutnya.

Nilai Mean RR sangat berhubungan sekali dengan nilai BPM. Karena rumus BPM adalah $60/\text{Mean RR}$. Ada batasan seseorang bisa dikatakan memiliki nilai BPM normal atau tidak, yaitu jika nilai BPM 60 - 100. Berarti Mean RR (atau rata-rata RR) paling minimal agar nilai BPM bisa disebut normal adalah 1 detik. Nilai normal Mean RR ada pada *range* 0,6 – 1,2 detik [27]. Seperti pada data yang di gunakan ini Mean RR nilainya 0,9 detik (hampir mendekati 1 detik). Tetapi nilai Mean RR dibawah 1 detik pun terbilang baik, dengan syarat masih di atas atau sama dengan nilai 0,6 detik. Yang perlu di waspadai adalah jika Mean RR lebih dari 1,2 detik (BPM 50) dan kurang dari 0,6 detik (BPM >100). Keadaan tersebut bisa mengindikasikan bahwa orang tersebut terkena aritmia [12].

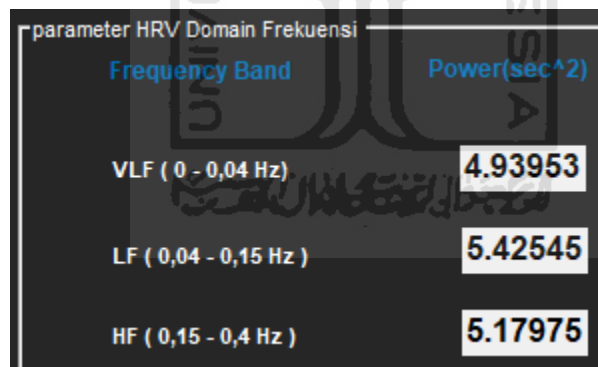
3. SDRR/SDNN

SDRR/SDNN atau standar deviasi RR interval. Standar deviasi atau simpangan baku (kuadrat varians) artinya secara umum adalah nilai rata-rata penyimpangan data, yang diukur dari nilai rata-rata data tersebut. Mudahnya adalah mengukur bagaimana nilai-nilai tersebar dalam suatu kelompok data. Semakin besar nilai SDRR, maka semakin jauh jarak rata-rata setiap unit data terhadap nilai mean data. Nilai SDRR/SDNN adalah suatu nilai yang penting pada HRV *domain* waktu. Karena nilai ini digunakan untuk menggolongkan secara medis subjek-subjek yang beresiko terkena penyakit kardiovaskular (data direkam selama 24 jam). Nilai ini juga digunakan untuk memprediksi morbitas dan mortalitas. Berdasarkan rekam data ECG 24 jam, pasien dengan nilai SDNN di bawah 50 ms dikategorikan sebagai tidak sehat, 50-100 ms cukup membahayakan kesehatannya, dan di atas 100 ms adalah sehat [1][28].

4. CVRR

CVRR atau rasio antara SDRR dengan Mean RR. Salah satu fungsi mengetahui nilai CVRR adalah nilai CVRR bisa menjadi indikator yang berpotensi bermanfaat untuk mengecek fungsi kardiovaskular pada individu sehat dan pasien stroke [23].

4.1.2 HRV *domain* frekuensi



Gambar 4. 5 HRV *domain* frekuensi

Dari ketiga pita frekuensi, nilai *power* LF adalah nilai terbesar. Ini menandakan bahwa sistem saraf simpatis dari subjek (yang datanya sedang di analisis) sedang mendominasi (lebih aktif) dibanding sistem saraf parasimpatisnya.

Sistem saraf simpatis umumnya berfungsi untuk merangsang kerja organ, seperti mempercepat detak jantung (meningkatkan BPM), menyebabkan kontraksi pembuluh darah, dan meningkatkan adrenalin. Dapat disimpulkan kemungkinan subjek sedang dalam keadaan aktif/bergerak, bukan sedang dalam keadaan santai/istirahat.

1. *Very Low Frequency (VLF)*

VLF adalah frekuensi yang berada pada pita frekuensi 0 – 0,04 Hz. *Power VLF* lebih kuat keterkaitannya sebagai penyebab kematian, daripada LF atau HF. Terbukti, *power VLF* yang rendah dikaitkan dengan kematian akibat aritmia dan *posttraumatic stress disorder (PTSD)*/gangguan stres pasca trauma. PTSD sendiri adalah kondisi psikologis seseorang yang dipicu oleh peristiwa traumatis masa lalu yang dirasakan sendiri sendiri atau hanya menyaksikannya [29][30]. *Power VLF* yang rendah dikorelasikan dengan rendahnya kadar testosteron. Ada ketidakpastian mengenai mekanisme fisiologis yang bertanggung jawab untuk aktivitas dalam pita ini. Sistem saraf intrinsik jantung berkontribusi pada ritme VLF dan aktivitas saraf simpatis, karena aktivitas fisik dan respons stres memodulasi amplitudo dan frekuensinya [28].

2. *Low Frequency (LF)*

Dalam kondisi istirahat, pita LF lebih mencerminkan aktivitas *baroreflex* dan bukan persarafan simpatis jantung [28]. *Baroreflex* atau refleksi baroreseptor adalah salah satu mekanisme *homeostatis* tubuh yang membantu menjaga tekanan darah pada tingkat yang hampir konstan [31].

Power LF dapat diproduksi oleh saraf simpatis dan parasimpatis. Tetapi lebih banyak *power LF* diaktifkan oleh aktifitas *baroreflex* [32]. Saraf simpatis menghasilkan ritme tidak jauh di atas 0,1 Hz itu menandakan jika sistem saraf simpatis lebih banyak bekerja. Sedangkan saraf parasimpatis mempengaruhi ritme sekitar 0,05 Hz [28].

Selama periode respirasi yang lambat, aktivitas vagal dapat dengan mudah menghasilkan osilasi dalam irama jantung yang melintas ke pita LF. Ketika seseorang menghela napas atau menarik napas dalam-dalam ini akan masuk pada pita LF [28].

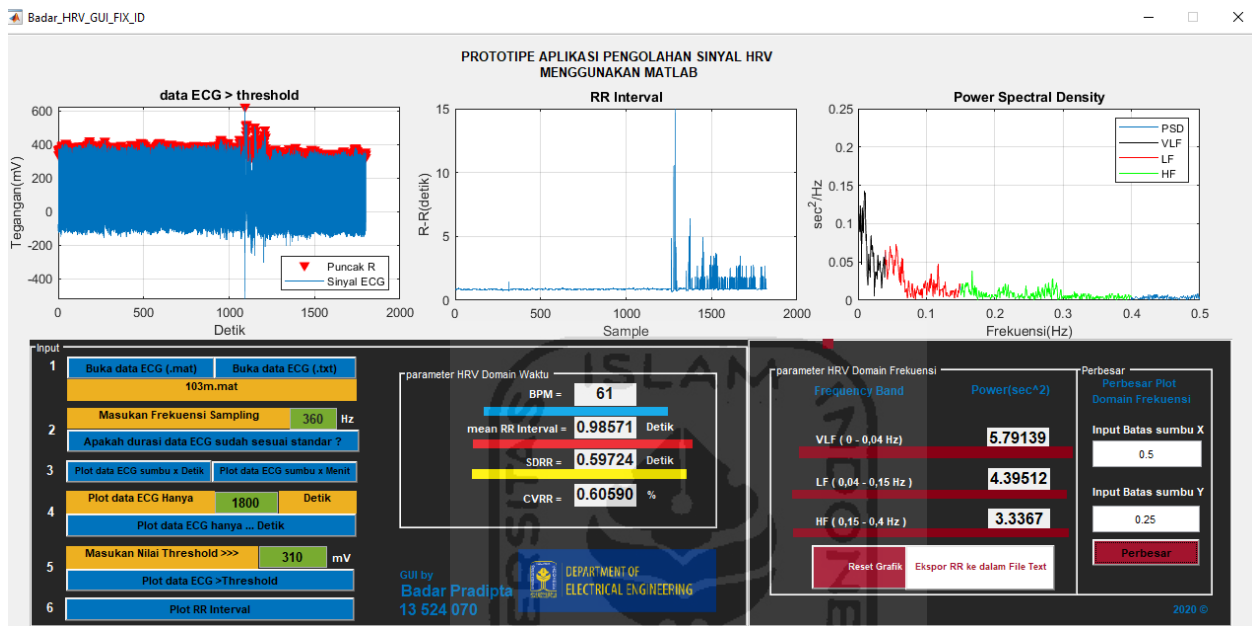
3. *High Frequency (HF)*

HF adalah frekuensi yang berada pada pita frekuensi 0,15 – 0,4 Hz. Pita HF mencerminkan aktivitas parasimpatis dan disebut pita pernapasan karena sesuai dengan variasi *heart rate* yang terkait dengan siklus pernapasan. Secara konvensional direkam selama periode minimum 1 menit, untuk bayi dan anak-anak yang bernapas lebih cepat daripada orang dewasa.

Power HF mungkin meningkat di malam hari dan menurun di siang hari. *Power HF* yang rendah berhubungan dengan stres, panik, cemas, atau khawatir. Modulasi nada vagal membantu menjaga regulasi otonom dinamis yang penting untuk kesehatan jantung. Kurangnya penghambatan vagal berimplikasi pada peningkatan morbiditas [28].

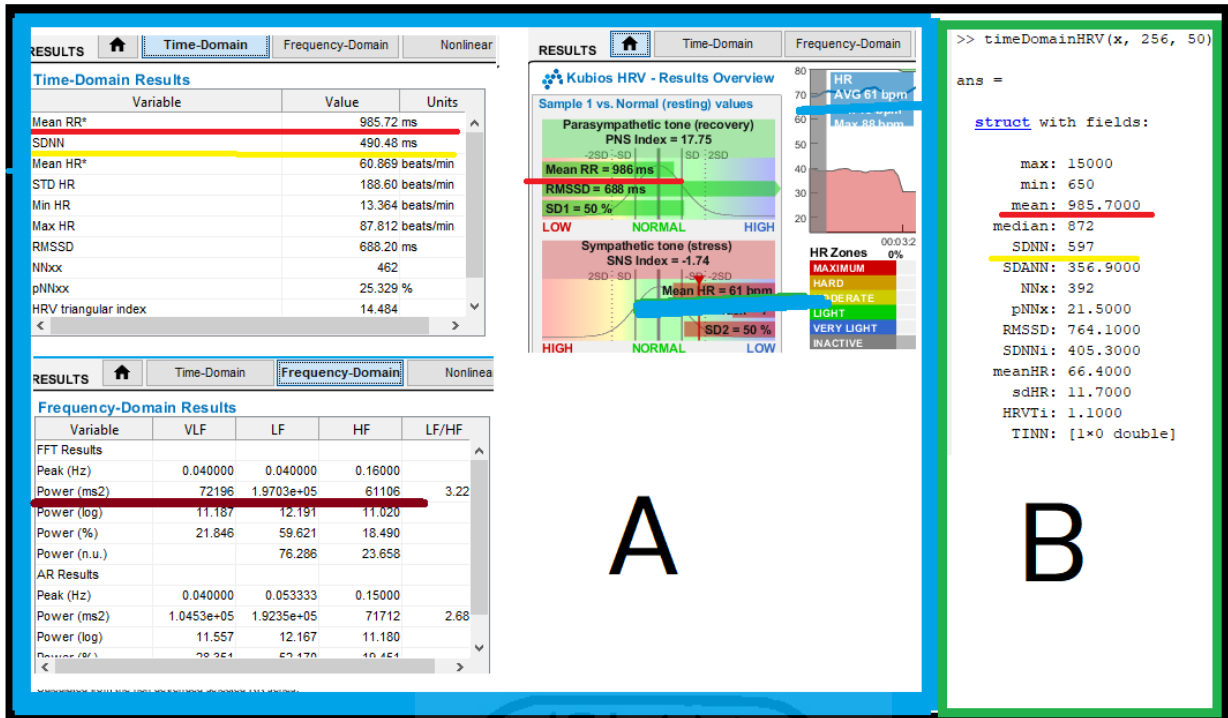
4.2 Validasi data

Output dari GUI akan coba dibandingkan dengan dua *tools* serupa (analisis HRV), yaitu *software* Kubios HRV dan Program Dari Internet (PDI). Data pertama yang akan di bandingkan adalah data dari MIT-BIH Arrhythmia database (*mitdb*) dengan nama file *file* 103m.mat. Frekuensi *sampling* 360 Hz, menggunakan nilai *threshold* 310 mV. Pada gambar 4.6 adalah hasil dari eksekusi program.



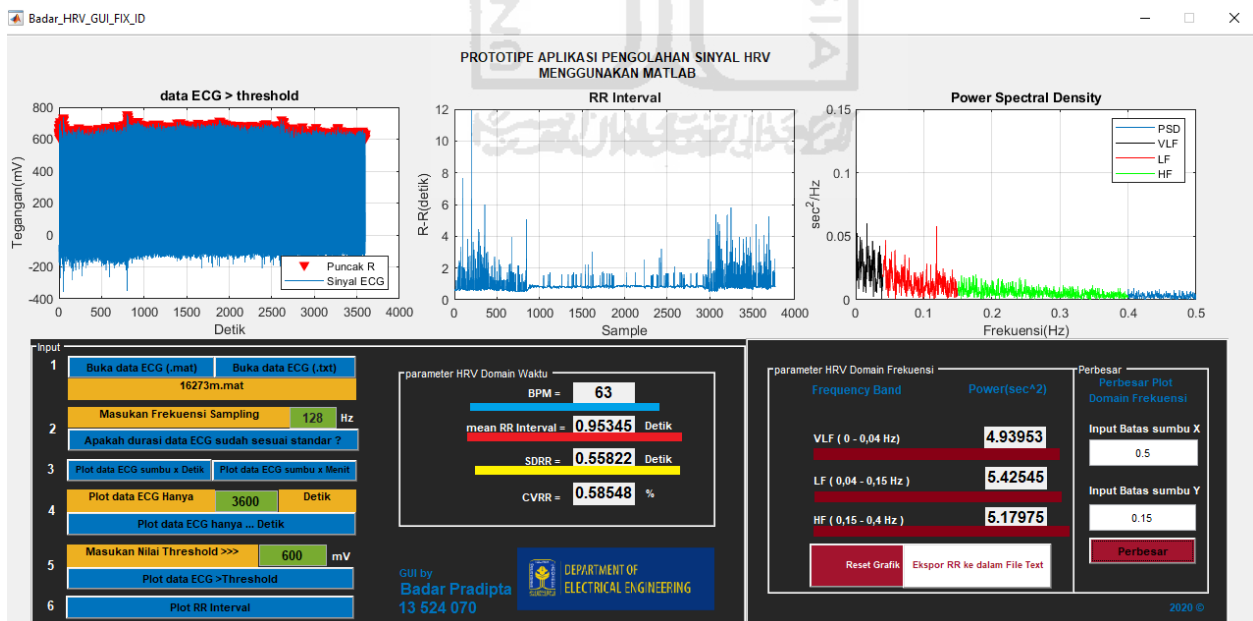
Gambar 4. 6 *Output* dari GUI 1 (*file* name 103m.mat)

Selanjutnya pada gambar 4.7 adalah *output* dari *software* Kubios HRV(gambar A-kiri) dan PDI (gambar B-kanan) dengan *file* ECG yang sama. Hasil dari ketiganya ditandai dengan warna garis bawah pada setiap parameternya. BPM dengan warna garis bawah biru. Mean RR dengan warna garis bawah merah. SDRR dengan warna garis bawah kuning. Untuk CVRR penulis buat perbandingan antara ketiganya dengan RR perhitungan manual. Dan untuk *power* VLF, LF, dan HF dengan warna garis bawah coklat.



Gambar 4. 7 Output Kubios HRV dan PDI 1 (file name 103m.mat)

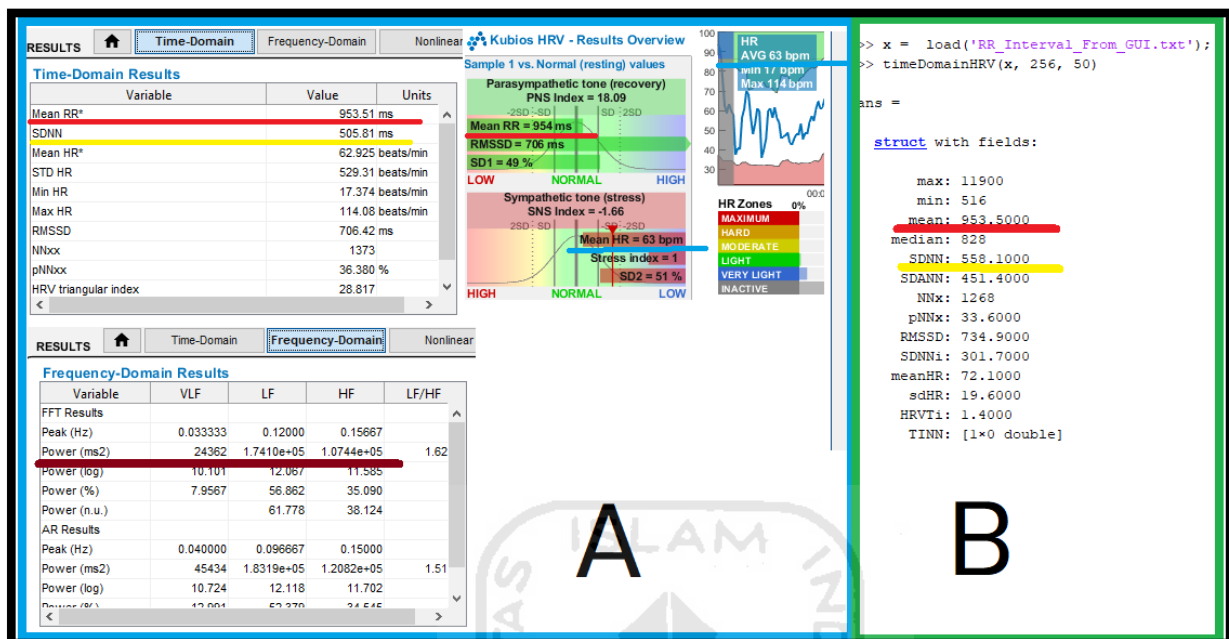
Data kedua yang akan dibandingkan adalah data dari MIT-BIH Normal Sinus Rhythm Database (nsrdb) dengan nama file 16273m.mat. Frekuensi sampling 128 Hz, menggunakan nilai threshold 600 mV. Pada gambar 4.8 adalah hasil dari GUI



Gambar 4. 8 Output dari GUI 2 (file name 16273m.mat)

Selanjutnya pada gambar 4.9 adalah output dari software Kubios HRV(gambar A-kiri) dan PDI (gambar B-kanan) dengan file ECG yang sama. Hasil dari ketiganya ditandai dengan warna

garis bawah pada setiap parameteranya. BPM dengan warna garis bawah biru. Mean RR dengan warna garis bawah merah. SDRR dengan warna garis bawah kuning. Untuk CVRR penulis buat perbandingan antara ketiganya dengan perhitungan manual. Dan untuk *power* VLF, LF, dan HF dengan warna garis bawah coklat.



Gambar 4. 9 Output Kubios HRV dan PDI 2 (file name 16273m.mat)

Penulis menggunakan *raw data* ECG untuk membandingkan *output* antara GUI, Kubios HRV, dan PDI. Dengan cara seperti inilah penulis melakukan perbandingan nilai antara ketiga *tools* ini. Penulis menggunakan 8 sampel data (4 data dengan durasi 30 menit dan 4 data dengan durasi 1 jam) *raw* ECG. Agar lebih mudah untuk dipahami, penulis membuat perbandingan dari 8 sampel data yang digunakan menggunakan table. Lebih jelasnya bisa dilihat pada tabel 4.1.

Tabel 4.1 Perbandingan hasil perhitungan parameter HRV

DATA ECG DURASI 30 MENIT								
Nama File ECG (.mat)	HASIL DARI	PARAMETER HRV DOMAIN WAKTU						
		MeanRR (ms)	SDRR (ms)	CVRR (%)	BPM	VLF (ms ²)	LF (ms ²)	HF (ms ²)
103m (T=310 mV)	GUI	985,771	597,24	0,60590	61	5791,39	4395,12	3336,7
	KUBIOS	985,72	490,48	0,49758	61	72196	197030	61106
	PDI	985,7	597	0,60566	61	-	-	-
234m. (T=260 mV)	GUI	706,9	464,93	0,65767	85	3423,48	4417,08	3653,11
	KUBIOS	706,96	503,70	0,712487	85	28695	286250	103280
	PDI	706,9	465	0,6578	85	-	-	-
104m T=120 mV	GUI	686,04	300,2	0,43758	87	1187,3	1074,66	1892,73
	KUBIOS	685,99	682,55	0,99498	87	72777	1340300	2809400
	PDI	686	300,2	0,4376	87	-	-	-
111m T=80 mV	GUI	659,26	718,49	1,08984	91	4487,93	4980,31	4944,83
	KUBIOS	659,27	913,95	1,3863	91	90283	1289600	3161500
	PDI	659,3	718,5	1,08979	91	-	-	-
DATA ECG DURASI 1 JAM								
16273m (T=600 mV)	GUI	953,45	558,22	0,58548	63	4939,53	5425,45	5179,75
	KUBIOS	953,51	505,81	0,53047	63	24362	174100	107440
	PDI	953,5	558,1	0,5853	63	-	-	-
16773m (T=400 mV)	GUI	808,47	140,12	0,17331	74	1391,2	653,436	818,901
	KUBIOS	808,53	106,06	0,13118	74	350,60	2783,3	3330,3
	PDI	808,5	140,2	0,1734	74	-	-	-
17453m (T=410 mV)	GUI	730,86	97,07	0,13282	82	1188,69	645,672	896,72
	KUBIOS	730,91	73,612	0,10071	82	265,57	2462,3	2994,5
	PDI	730,9	97,1	0,1328	82	-	-	-
18184m (T=200 mV)	GUI	673,87	137,32	0,20377	89	1163,67	465,11	519,68
	KUBIOS	674	91,248	0,13538	89	151,63	1293,6	628,18
	PDI	673,9	137,3	0,203591	89	-	-	-

Keterangan :

T = Nilai *threshold* yang digunakan

Ada beberapa catatan penting dari tabel 4.1 ini yaitu :

- Satuan asli Mean RR, SDRR dan *power* pada setiap pita frekuensi pada GUI adalah *second(sec)*. Tetapi dalam tabel satuan dibuat menjadi *millisecond (ms)* agar seragam dengan Kubios HRV dan PDI. Serta agar mudah untuk mencari nilai selisih diantara ketiganya.
- Dilakukan pembulatan pada nilai *power* 3 pita frekuensi (VLF, LF dan HF) yang dirubah dari satuan *second(sec)* ke *millisecond(ms)*.
- Perhitungan perbandingan/validasi data dilakukan dengan metode validasi seperti yang dijelaskan pada sub bab 3.4.

Hasil dari pembacaan tabel sebagai berikut :

1. Nilai BPM pada ketiga *tools* tersebut setidaknya ada selisih, yang artinya perhitungan BPM pada program GUI sudah benar.

2. MeanRR

- Untuk data ECG dengan durasi 30 menit dengan nama *file*

103m.mat

GUI – Kubios HRV

985,771 - 985,72 = 0,699 ms

GUI – PDI

985,771 – 985,7 = 0,071 ms.

234m..mat

GUI – Kubios HRV

706,9- 706,96= 0,06 ms

GUI – PDI

706,9 – 706,9 = 0 ms

104m.mat

GUI – Kubios HRV

686,04 - 685,99 = 0,05 ms

GUI – PDI

686,04 – 686 = 0,04 ms.

111m..mat

GUI – Kubios HRV

659,26 - 659,27 = 0,01 ms

GUI – PDI

659,26 - 659,3= 0,04 ms

- Untuk data ECG dengan durasi 1 jam dengan nama *file*

16273m.mat

GUI – Kubios HRV

953,45- 953,51 = 0,06 ms

GUI – PDI

953,45 - 953,5 = 0,05 ms

16773m..mat

GUI – Kubios HRV

808,47 - 808,53 = 0,06 ms

GUI – PDI

808,47 - 808,5 = 0,03 ms

17453m.mat
GUI – Kubios HRV
 $730,86 - 730,91 = 0,05$ ms
GUI – PDI
 $730,86 - 730,9 = 0,04$ ms

18184m..mat
GUI – Kubios HRV
 $673,87 - 674 = 0,13$ ms
GUI – PDI
 $673,87 - 673,9 = 0,03$ ms

Selisih rata-rata Mean RR antara GUI dan Kubios HRV adalah

$(0,699+0,06+0,05+0,01+0,06+0,06+0,05+0,13) / 8$ (sampel) = $1,119/8 = \mathbf{0,1398}$ ms

Selisih rata-rata Mean RR antara GUI dan PDI adalah

$(0,071+0+0,04+0,04+0,05+0,03+0,04+0,03) / 8$ (sampel) = $0,301/8 = \mathbf{0,0376}$ ms

Dari 8 sampel data ECG yang digunakan, nilai selisih rata-rata Mean RR paling besar berasal dari perbandingan dengan Kubios HRV yaitu 0,1398 ms/0,0001398 sec. Karena Mean RR adalah nilai tengah dari suatu kelompok data RR interval, memang sudah seharusnya antara GUI dan 2 *tools* lainnya tidak memiliki selisih nilai yang terlalu besar. Dari validasi ini dapat disimpulkan perhitungan Mean RR pada program GUI sudah benar.

3. SDRR

- Untuk data ECG dengan durasi 30 menit dengan nama *file*

103m.mat
GUI – Kubios HRV
 $597,24 - 490,48 = 106,76$ ms
GUI – PDI
 $597,24 - 597 = 0,24$ ms.

234m..mat
GUI – Kubios HRV
 $464,93 - 503,70 = 38,77$ ms
GUI – PDI
 $464,93 - 465 = 0,07$ ms

104m.mat
GUI – Kubios HRV
 $300,2 - 682,55 = 382,35$ ms
GUI – PDI
 $300,2 - 300,2 = 0$ ms.

111m..mat
GUI – Kubios HRV
 $718,49 - 913,95 = 195,46$ ms
GUI – PDI
 $718,49 - 718,5 = 0,44$ ms

- Untuk data ECG dengan durasi 1 jam dengan nama *file*

16273m.mat

GUI – Kubios HRV

558,22 – 505,81 = 52,41 ms

GUI – PDI

558,22 - 558,1 = 0,21 ms.

16773m..mat

GUI – Kubios HRV

140,12 – 106,06 = 34,06 ms

GUI – PDI

140,12 – 140,2 = 0,10 ms

17453m.mat

GUI – Kubios HRV

97,07 – 73,612 = 23,458 ms

GUI – PDI

97,07 – 97,1 = 0,03 ms.

18184m..mat

GUI – Kubios HRV

137,32 – 91,248 = 46,072 ms

GUI – PDI

137,32 – 137,2 = 0,30 ms

Selisih rata-rata SDRR antara GUI dan Kubios HRV adalah

$(106,76+38,77+52,41+34,06+382,35+195,46+23,458+46,072) / 8$ (sampel) = $879,34/8 =$
109,9175 ms

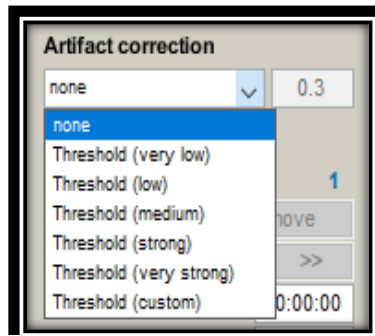
Selisih rata-rata SDRR antara GUI dan PDI adalah

$(0,24+0,07+0+0,44+0,21+0,10+0,03+0,3) / 8$ (sampel) = $1,39/8 =$ **0,17375** ms

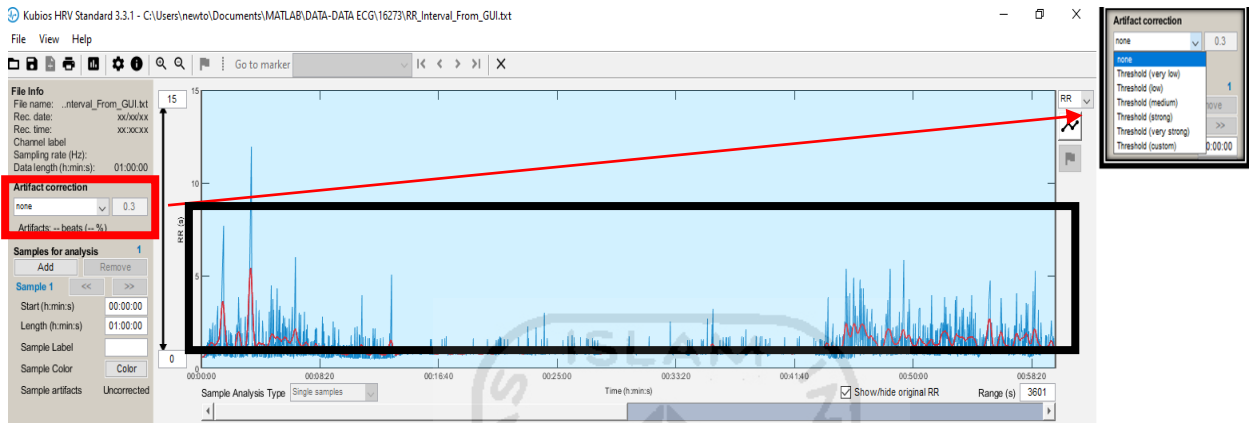
Dari 8 sampel data ECG yang digunakan, selisih nilai rata-rata SDRR paling besar berasal dari perbandingan dengan Kubios HRV yaitu 109,9175 ms/0,1099175 sec. Nilai SDRR yang normal adalah diatas 100 ms [1][28]. Jika selisihnya mencapai 100 ms keatas seperti pada Kubios HRV, ada kemungkinan ketika nilai pada GUI sudah di katakan normal, sedangkan nilai pada Kubios HRV belum bisa di katakan (mencapai batas) normal.

Misalnya, nilai SDRR pada GUI 150 ms. Kubios HRV 40 ms. Bisa dikatakan bahwa nilai SDRR GUI sudah normal sedangkan Kubios HRV belum. Ini terjadi karena ada selisih nilai sebesar 109,9175 ms (seperti pada nilai validasi).

Selisih nilai yang cukup besar ini terjadi karena Kubios HRV yang digunakan untuk memvalidasi mempunyai fitur *artifact correction*, yang berfungsi untuk mengurangi (membuat normal) nilai RR Interval yang tidak normal. Fitur *artifact correction* bisa dilihat pada gambar 4.10 dan contoh hasil dari *artifact correction* bisa dilihat pada gambar 4.11.



Gambar 4. 10 *Artifact correction*



Gambar 4. 11 Contoh *artifact correction* pada Kubios HRV

Fitur *artifact correction* pada Kubios HRV berperan seperti *threshold* pada GUI. Fitur ini memiliki beberapa pilihan seperti *none* (*default*), *very low*, *low*, *medium*, *strong*, *very strong*, dan *custom*. Penulis menggunakan *artifact correction* jenis *none* (*default*). Jika ada RR Interval yang tidak normal, maka akan dijadikan nilai normal oleh fitur ini (sesuai jenis *artifact correction* yang dipilih). Dan setiap pilihan akan menghasilkan karakteristik HRV yang berbeda. Misalnya, ada nilai RR Interval yang lebih dari 5 detik, Kubios HRV dengan *artifact correction*nya akan menjadikan nilai RR Interval tersebut masuk ke dalam *range* normal (sesuai dengan jenis *artifact correction* yang dipilih) kembali. Itulah mengapa nilai SDRR antara GUI dan Kubios HRV berbeda (walaupun *input* Kubios HRV adalah data RR Interval dari GUI).

Selain berpengaruh pada nilai SDRR, *artifact correction* juga akan mempengaruhi nilai *power* dan urutan pola pita di setiap frekuensi (VLF, LF, dan HF) pada HRV *domain* frekuensi.

4. CVRR

Karena nilai CVRR tidak ada pembandingnya, penulis akan menghitung CVRR secara manual. CVRR dapat dihitung dengan rumus = $(\text{MeanRR}/\text{SDRR}) \times 100\%$. Perhitungan manual yang penulis buat sebagai berikut :

- Untuk data ECG dengan durasi 30 menit dengan nama *file*

103m.mat

- GUI = 0,60590 %
 - Kubios HRV =
 $490,48 / 985,72 = 0,49758 \%$
 > GUI – Kubios HRV
 $0,60590 - 0,49758 = \mathbf{0,10832 \%$

- PDI =
 $597 / 985,7 = 0,60566 \%$
 > GUI – PDI
 $0,60590 - 0,60566 = \mathbf{0,00024 \%$

104m.mat

- GUI = 0,43758 %
 - Kubios HRV =
 $682,55 / 685,99 = 0,99498 \%$
 > GUI – Kubios HRV
 $0,43758 - 0,99498 = \mathbf{0,5574 \%$

- PDI =
 $300,2 / 686 = 0,4376 \%$
 > GUI – PDI
 $0,43758 - 0,4376 = \mathbf{0,00002 \%$

234m.mat

- GUI = 0,65767 %
 - Kubios HRV =
 $503,70 / 706,96 = 0,712487 \%$
 > GUI – Kubios HRV
 $0,65767 - 0,712487 = \mathbf{0,054817 \%$

- PDI =
 $465 / 706,9 = 0,6578 \%$
 > GUI – PDI
 $0,65767 - 0,6578 = \mathbf{0,00013 \%$

111m.mat

- GUI = 1,08984 %
 - Kubios HRV =
 $913,95 / 659,27 = 1,3863 \%$
 > GUI – Kubios HRV
 $1,08984 - 1,3863 = \mathbf{0,29646 \%$

- PDI =
 $718,5 / 659,3 = 1,08979 \%$
 > GUI – PDI
 $1,08984 - 1,08979 = \mathbf{0,00005 \%$

- Untuk data ECG dengan durasi 1 jam dengan nama *file*

16273m.mat

- GUI = 0,58548 %
 - Kubios HRV =
 $505,81 / 953,51 = 0,53047 \%$
 > GUI – Kubios HRV
 $0,58548 - 0,53047 = \mathbf{0,05501 \%$

- PDI =
 $558,1 / 953,5 = 0,5853 \%$
 > GUI – PDI
 $0,58548 - 0,5853 = \mathbf{0,00018 \%$

16773m.mat

- GUI = 0,17331 %
 - Kubios HRV =
 $106,06 / 808,53 = 0,13118 \%$
 > GUI – Kubios HRV
 $0,17331 - 0,13118 = \mathbf{0,04213 \%$

- PDI =
 $140,2 / 808,5 = 0,1734 \%$
 > GUI – PDI
 $0,17331 - 0,1734 = \mathbf{0,00009 \%$

17453m.mat

- GUI = 0,13282 %
- Kubios HRV =
73,612 / 730,91 = 0,10071 %
> GUI – Kubios HRV
0,13282 - 0,10071 = **0,03211** %

- PDI
97,1 / 730,9 = 0,1328 %
> GUI – PDI
0,13282 - 0,1328 = **0,00002** %

18184m.mat

- GUI = 0,20377 %
- Kubios HRV =
91,248 / 674 = 0,13538 %
> GUI – Kubios HRV
0,20377 - 0,13538 = **0,06839** %

- PDI
137,2 / 673,9 = 0,203591 %
> GUI – PDI
0,20377 - 0,203591 = **0,000179** %

Selisih rata-rata CVRR antara GUI dan Kubios HRV adalah

$$(0,10832+0,054817+0,5574+0,29646+0,05501+0,04213+0,03211+0,06839) / 8 \text{ (sampel)}$$
$$= 1,214637/8 = \mathbf{0,151829625} \%$$

Selisih rata-rata CVRR antara GUI dan PDI adalah

$$(0,00024+0,00013+0,00002+0,00005+0,00018+0,00009+0,00002+0,000179)/8 \text{ (sampel)}$$
$$= 0,000909/8 = \mathbf{0,000113625} \%$$

Nilai CVRR sangat bergantung dengan nilai SDRR dan Mean RR. Nilai CVRR sendiri dalam bentuk presentase.

Selisih nilai CVRR paling besar ada pada perbandingan dengan Kubios yaitu, 0,151829625. Sedangkan perbandingan dengan PDI, selisih nilainya hanya 0,000113625. Dari validasi ini dengan 2 *tools* ini dapat disimpulkan perhitungan CVRR sudah benar, karena perhitungan SDRR dan Mean RR juga benar.

Seperti yang sudah dijelaskan pada halaman 36, terjadi perbedaan nilai yang sangat besar pada nilai *power* dan pada pola urutan setiap pita frekuensinya (VLF, LF, dan HF) antara GUI dan Kubios HRV. Karena Kubios HRV menggunakan fitur *artifact correction*, yang berfungsi untuk mengurangi (membuat normal) nilai RR Interval yang tidak normal.

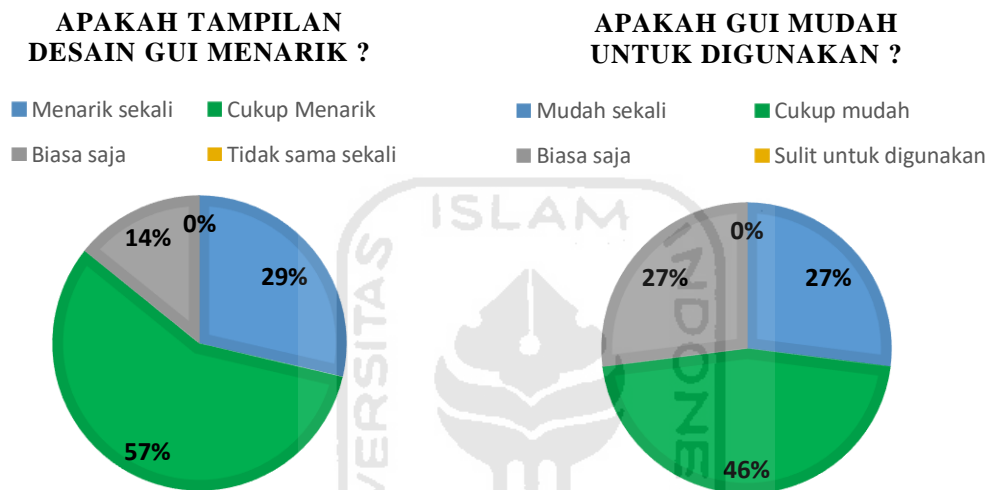
Dari 8 sampel (4 data durasi 30 menit dan 4 data durasi 1 jam) data ECG yang dibandingkan hanya ada 2 sampel data yang pola pita frekuensinya sama. Yaitu data ECG dengan nama *file* 234m.mat (30 menit) dan 16273m.mat (1 jam). Setiap pilihan (jenis) *artifact correction* akan menghasilkan karakteristik nilai parameter HRV *domain* frekuensi yang berbeda-beda.

4.3 User experience

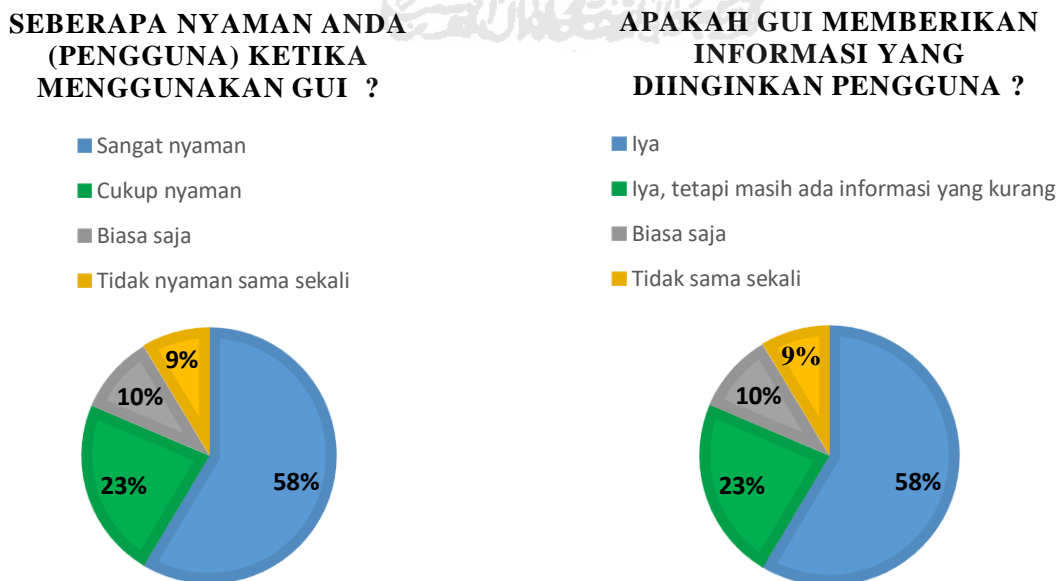
Dilakukan evaluasi *user experience* dengan memberikan *form online* dengan beberapa pertanyaan yang dijadikan sebagai parameter penilaian seperti kenyamanan, kemudahan, dan

kepuasan *user* saat menggunakan GUI. Subjek yang digunakan untuk *user experience* ada 7 orang, yang semuanya adalah mahasiswa (dan alumni) Teknik Elektro UII yang terbiasa menggunakan program MATLAB. *Raw data* ECG yang mereka gunakan adalah *raw data* ECG yang penulis pakai dalam penelitian ini.

Diberikan 5 pertanyaan yang berhubungan dengan kepuasan *user* terhadap GUI. Setiap pertanyaan memiliki 4 pilihan jawaban, yang setiap jawabannya memiliki nilai masing-masing. Jawaban ke 1 memiliki nilai 10, jawaban ke 2 memiliki nilai 7,5, jawaban ke 3 memiliki nilai 5 dan jawaban ke 4 memiliki nilai 2,5. Jawaban ditampilkan dalam nilai persentase. Berikut adalah beberapa pertanyaan yang penulis ajukan kepada *user* beserta jawabannya :



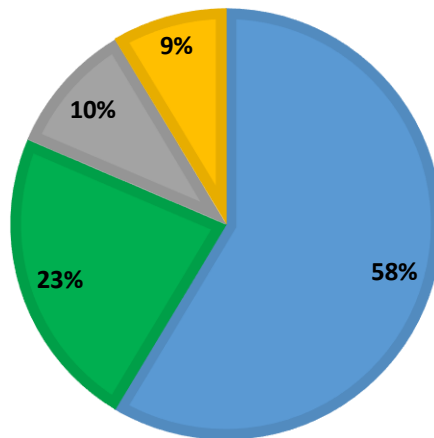
Gambar 4. 12 Hasil pertanyaan *user experience* 1 dan 2



Gambar 4. 13 Hasil pertanyaan *user experience* 3 dan 4

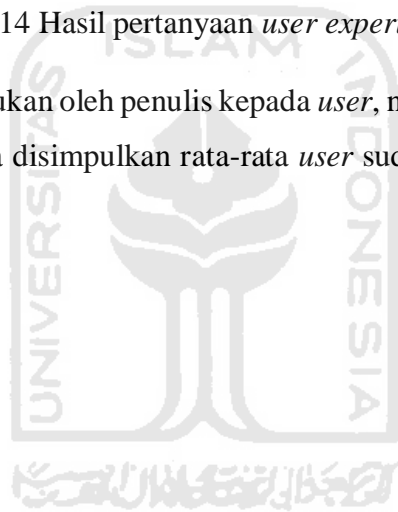
**APAKAH ANDA (PENGGUNA)
PUAS DNEGAN GUI INI ?**

- Sangat puas
- Cukup puas
- Biasa saja
- Tidak sama sekali



Gambar 4. 14 Hasil pertanyaan *user experience* 5

Dari 5 pertanyaan yang diajukan oleh penulis kepada *user*, nilai *performance* rata-rata yang diberikan GUI adalah 55,4 %. Bisa disimpulkan rata-rata *user* sudah merasa cukup puas dengan apa yang diberikan oleh GUI.



BAB 5

KESIMPULAN DAN SARAN

5.1 Kesimpulan

1. GUI yang telah dirancang menunjukkan hasil HRV *domain* waktu dan *domain* frekuensi yang memiliki karakteristik yang tidak jauh berbeda dengan *free analysis software* HRV yang sudah ada. Contohnya, nilai BPM antara GUI dengan Kubios HRV tidak ada selisih nilai sama sekali. Ini menunjukkan perhitungan pada program GUI sudah benar.
2. Selisih perbandingan rata-rata nilai terbesar pada :
 - Mean RR, yaitu 0,1398 ms.
 - SDRR, yaitu 109,9175 ms. Selisih nilai yang cukup besar ini terjadi karena pada Kubios HRV terdapat fitur *artifact correction*, yang akan mengoreksi nilai RR Interval yang tidak normal.
 - CVRR, yaitu 0,1518%.

Dapat diambil kesimpulan perhitungan pada program GUI sudah relatif benar.

3. Untuk HRV *domain* frekuensi terjadi perbedaan nilai *power* dan pola urutan pita frekuensi, karena pada Kubios HRV terdapat fitur *artifact correction*. Yang mana pada setiap jenis *artifact correction* yang dipilih, akan menghasikan karakteristik nilai parameter HRV *domain* frekuensi yang berbeda-beda.

5.2 Saran

1. Agar GUI kedepannya bisa menganalisis *format file* ECG seperti .dat, .csv, dll.
2. Penambaham proses filter pada GUI dan metode analisi HRV lain seperti *nonlinear analysis*.

DAFTAR PUSTAKA

- [1] U. Indonesia et al., Kemampuan heart rate variability metode photoplethysmografi sebagai prediktor major adverse cardiac events pada pasien sindrom koroner akut metode photoplethysmografi sebagai. 2015.
- [2] P. Novani, “Heart rate variability frekuensi domain untuk deteksi stres mental dan influenza menggunakan SVM classifier,” vol. 9, no. 2, hal. 88, 2016.
- [3] D. Guspriyadi, J. T. Industri, H. R. Variability, dan T. D. Analysis, “Analisis tingkat stres dan tingkat kelelahan masinis berdasarkan heart rate variability,” vol. 02, no. 02, hal. 57–67, 2014.
- [4] R. M. Soleh, A. Rizal, dan R. Magdalena, “Menggunakan algoritma iterative threshold pada subband wavelet denoising electrocardiogram (ECG) recording using iterative threshold at wavelet subband otot jantung . EKG ini merupakan rekaman informasi kondisi jantung yang diambil menggunakan.” hal. 1–6, 2008.
- [5] H. F. Jelinek, D. J. Cornforth, dan A. H. Khandoker, “Introduction to ECG time series variability analysis: A simple overview,” ECG Time Ser. Var. Anal. Eng. Med., hal. 1–12, 2017, doi: 10.4324/9781315372921.
- [6] J. Halomoan, “Analisis Sinyal EKG dengan Metoda HRV (Heart Rate Variability) pada Domain Waktu Aktivitas Berdiri dan Terlentang,” Semin. Nas. Apl. Teknol. Inf. 2013, hal. 29–35, 2013.
- [7] T. L. Hartono, F. D. Setiaji, I. Setyawan, P. Studi, T. Elektro, dan F. Teknik, “Alat bantu analisis,” Techné J. Ilm. Elektrotek., vol. 12 No. 2 O, hal. 141–158, 2013.
- [8] A. W. Setiawan, R. A. Djohan, dan F. I. Tawakal, “Deteksi Aritmia Menggunakan Sinyal EKG dengan Metoda Deteksi Puncak-R,” Seniati, vol. 5, hal. 123–128, 2019.
- [9] Wikipedia, “Bundle_of_His.” [Daring]. Tersedia pada: https://en.wikipedia.org/wiki/Bundle_of_His#Function.
- [10] D. Benítez, D. Mascher, dan J. Alanis, “The electrical activity of the bundle of His - The fast and slow inward currents,” Pflügers Arch. Eur. J. Physiol., vol. 345, no. 1, hal. 61–72, 1973, doi: 10.1007/BF00587062.
- [11] D. Meningioma, D. A. N. Schwannoma, D. Citra, M. Gray, dan L. C. Matrices, “Adln – perpustakaan universitas airlangga deteksi,” no. Bagian Ii, hal. 37–42, 2016.
- [12] S. Nadler, “Heart rhythms,” Mass. Rev., vol. 42, no. 2, hal. 209–222, 2002.
- [13] <https://www.heart.org/>, “All About Heart Rate (Pulse).” [Daring]. Tersedia pada: <https://www.heart.org/en/health-topics/high-blood-pressure/the-facts-about-high-blood-pressure/all-about-heart-rate-pulse>.
- [14] D. M. S. H. A. Permana, “Elektrokardiograf (ekg) berbasis bluetooth,” Fis. Fak. Sains Teknol. , UIN Sunan Gunung Jati Bandung, vol. 2, no. 1, hal. 38–46, 2015.
- [15] I. Purnawan, “Konsep Dasar EKG,” hal. 2–3, 2014.
- [16] J. Arifin dan U. Elisa, Elektronika Biomedik, Pembahasan Sinyal Dan Citra

Elektrokardiografi (EKG).

- [17] Jaenal Arifin., “Identifikasi dan Klasifikasi Pola Sinyal EKG Berdasarkan Sifat Keacakan (Entropy),” 5th Natl. Conf. Inf. Technol. Electr. Eng. 2013, Yogyakarta, , vol. 5, no. 5, hal. 5, 2013.
- [18] B. Cardiac dan L. Support, “EKG Konsep Dasar”
- [19] U. R. Acharya, K. P. Joseph, N. Kannathal, C. M. Lim, dan J. S. Suri, “Heart rate variability: A review,” *Med. Biol. Eng. Comput.*, vol. 44, no. 12, hal. 1031–1051, 2006, doi: 10.1007/s11517-006-0119-0.
- [20] Garmin.com, “Indeks Referensi Alternatif untuk Memantau Kelelahan – Variabilitas Denyut Jantung (HRV).” [Daring]. Tersedia pada: <https://www.garmin.com/id-ID/blog/indeks-referensi-alternatif-untuk-memantau-kelelahan-variabilitas-denyut-jantung-hrv/>.
- [21] J. Sztajzel, “Heart rate variability: A noninvasive electrocardiographic method to measure the autonomic nervous system,” *Swiss Med. Wkly.*, vol. 134, no. 35–36, hal. 514–522, 2004, doi: 2004/35/smw-10321.
- [22] D. Razanskaite-Virbickiene, E. Danyte, G. Mockeviciene, R. Dobrovolskiene, R. Verkauskiene, dan R. Zalinkevicius, “Can coefficient of variation of time-domain analysis be valuable for detecting cardiovascular autonomic neuropathy in young patients with type 1 diabetes: A case control study,” *BMC Cardiovasc. Disord.*, vol. 17, no. 1, hal. 1–9, 2017, doi: 10.1186/s12872-016-0467-0.
- [23] C.-C. J. Change, T.-C. R. Hsiao, dan H.-Y. Hsu, “The Usefulness of the Coefficient of Variation of Electrocardiographic RR Interval as an Index of Cardiovascular Function and its Correlation with Age and Stroke,” *Tungs’ Med. J.*, vol. 6, no. March, hal. 41–48, 2012.
- [24] D. Singh, K. Vinod, dan S. C. Saxena, “Sampling frequency of the RR interval time series for spectral analysis of heart rate variability,” *J. Med. Eng. Technol.*, vol. 28, no. 6, hal. 263–272, 2004, doi: 10.1080/03091900410001662350.
- [25] Gunggung, “Metode ‘Napas Ritme’ dan Imunitas Tubuh.” [Daring]. Tersedia pada: <https://www.wisatabugarofficial.com/2020/05/metode-napas-ritme-dan-imunitas-tubuh/>.
- [26] Alodokter dan K. Adrian, “Ketahui Alasan Detak Jantung Cepat dan Cara Mengatasinya.” [Daring]. Tersedia pada: <https://www.alodokter.com/ketahui-alasan-detak-jantung-cepat-dan-cara-mengatasinya>.
- [27] L. Rosenthal, E. A. Gracia, dan B. Christensen, “Normal Electrocardiography (ECG) Intervals.” [Daring]. Tersedia pada: [https://emedicine.medscape.com/article/2172196-overview#:~:text=Normal ECG values for waves,PR interval%3A 120-200 milliseconds](https://emedicine.medscape.com/article/2172196-overview#:~:text=Normal ECG values for waves,PR interval%3A 120-200 milliseconds.).
- [28] F. Shaffer dan J. P. Ginsberg, “An Overview of Heart Rate Variability Metrics and Norms,” *Front. Public Heal.*, vol. 5, no. September, hal. 1–17, 2017, doi: 10.3389/fpubh.2017.00258.
- [29] M. D. Cristy, “PTSD.”
- [30] Wikipedia, “Posttraumatic stress disorder.”
- [31] Wikipedia, “Baroreflex.” [Daring]. Tersedia pada: <https://en.wikipedia.org/wiki/Baroreflex>.
- [32] F. Rahman, S. Pechnik, D. Gross, L. T. Sewell, dan D. S. Goldstein, “Low frequency power

- of heart rate variability reflects baroreflex function, not cardiac sympathetic innervation,” *Clin. Auton. Res.*, vol. 21, no. 3, hal. 133–141, 2011, doi: 10.1007/s10286-010-0098-y.
- [33] D. Meningioma, D. A. N. Schwannoma, D. Citra, M. Gray, dan L. C. Matrices, “Adln perpustakaan universitas airlangga deteksi,” no. Bagian Ii, hal. 37–42, 2016.
- [34] [Http://dewara.com/](http://dewara.com/), “Cara Membaca EKG (Elektrokardiograf).” [Daring]. Tersedia pada: <http://dewara.com/cara-membaca-ekg-elektrokardiograf/>.
- [35] Pinterest, “signal complex pqrst.” [Daring]. Tersedia pada: <https://www.pinterest.com/pin/220887556710065555/>.
- [36] A. Murjana, “Rumus Interpolasi – Pengertian, Macam-Macam, Contoh Soal,” 2020. [Daring]. Tersedia pada: https://rumusrumus.com/rumus-interpolasi/#Interpolasi_Linear.
- [37] L. Yang, Q. Pan, G. Zhou, L. Fang, P. Cao, dan G. Ning, “Resampling the RR tachogram enhances the deceleration capacity of heart rate in the assessment of chronic heart failure,” *Proc. - 2017 10th Int. Congr. Image Signal Process. Biomed. Eng. Informatics, CISP-BMEI 2017*, vol. 2018-Janua, hal. 1–5, 2018, doi: 10.1109/CISP-BMEI.2017.8302234.



LAMPIRAN

```
%%%%%%%% ----- GUI BY : BADAR PRADIPTA -----
%%%%%%%% ----- contact : newtonbadar@gmail.com -----
%%%%%%%% LAST EDITED 20 AUGUST 2020
```

```
%%% -----
%%% ----- AWALAN PROGRAM DEFAUTL DARI MATLAB ----- %%%
%%% -----
```

```
function varargout = Badar_HRV_GUI_FIX_ID(varargin)
% BADAR_HRV_GUI_FIX_ID MATLAB code for Badar_HRV_GUI_FIX_ID.fig
%   BADAR_HRV_GUI_FIX_ID, by itself, creates a new BADAR_HRV_GUI_FIX_ID or
raises the existing
%   singleton*.
%
%   H = BADAR_HRV_GUI_FIX_ID returns the handle to a new
BADAR_HRV_GUI_FIX_ID or the handle to
%   the existing singleton*.
%
%   BADAR_HRV_GUI_FIX_ID('CALLBACK',hObject,eventData,handles,...) calls
the local
%   function named CALLBACK in BADAR_HRV_GUI_FIX_ID.M with the given input
arguments.
%
%   BADAR_HRV_GUI_FIX_ID('Property','Value',...) creates a new
BADAR_HRV_GUI_FIX_ID or raises the
%   existing singleton*. Starting from the left, property value pairs are
%   applied to the GUI before Badar_HRV_GUI_FIX_ID_OpeningFcn gets called.
An
%   unrecognized property name or invalid value makes property application
%   stop. All inputs are passed to Badar_HRV_GUI_FIX_ID_OpeningFcn via
varargin.
%
%   *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
%   instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help Badar_HRV_GUI_FIX_ID

% Last Modified by GUIDE v2.5 01-Jun-2020 15:10:14

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @Badar_HRV_GUI_FIX_ID_OpeningFcn, ...
                  'gui_OutputFcn',  @Badar_HRV_GUI_FIX_ID_OutputFcn, ...
                  'gui_LayoutFcn',  [], ...
                  'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
```

```

% End initialization code - DO NOT EDIT .

%%% -----
%%% ----- UNTUK MENAMPILKAN LOGO UII PADA GUI -----
function Badar_HRV_GUI_FIX_ID_OpeningFcn(hObject, eventdata, handles,
varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to Badar_HRV_GUI_FIX_ID (see VARARGIN)

% Choose default command line output for Badar_HRV_GUI_FIX_ID
handles.output = hObject;
axes(handles.axes8);
imshow('uui.png');

% Update handles structure
guidata(hObject, handles);
% UIWAIT makes Badar_HRV_GUI_FIX_ID wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = Badar_HRV_GUI_FIX_ID_OutputFcn(hObject, eventdata,
handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

%%% -----
%%% ----- AKHIRAN PROGRAM DEFAULTL DARI MATLAB -----   %%%
%%% -----
% % % % .....
% % % % .....
% % % % .....
% % % % .....
% % % % .....
% % % % .....
% % % % .....
% % % % .....
% % % % .....
% % % % .....
% % % % .....
% % % % .....
% % % % .....

%%% AXES 4 JANGAN DIHAPUS !!!
% --- Executes during object creation, after setting all properties.
function axes4_CreateFcn(hObject, eventdata, handles)
% hObject    handle to axes4 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
% Hint: place code in OpeningFcn to populate axes4

%%% -----
%%% ----- AWALAN FUNGSI UTAMA GUI -----
%%% -----

%%%%%%%% RAW DATA ECG BISA DI OLAH SAMPAI MENJADI HRV DOMAIN FREKUENSI
%%%%%%%% DENGAN SYARAT BARIS HARUS 1, DAN KOLOM TERSEERAH BEBAS

```



```

%%%%%%%%>>>>>>>>>> LANGKAH KE 3 (1)
%%%%%%%%>>>>> PLOT DALAM DETIK
% --- Executes on button press in InSec.
function inseccallback(hObject, eventdata, handles)

global A
global fs

global timeSec
timeSec = (0:numel(A)-1)/fs;
%%%%%%%%^
global Tmin
Tmin = fs*60*5;    %% Tmin = sample data dalam 5 menit

global P_Tmin
P_Tmin = length(A);
%%%%%%%%^
axes(handles.axes4);
hold off
plot(timeSec,A);
hold on
xlabel('Detik'), ylabel('Tegangan (mV)')
title('data ECG')
grid on
%%% =====
%%% =====

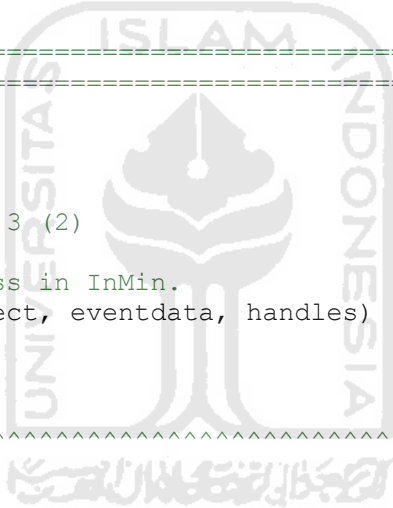
%%%%%%%%>>>>>>>>>> LANGKAH KE 3 (2)
%%%%%%%%>>>>> PLOT DALAM MENIT
% --- Executes on button press in InMin.
function inmincallback(hObject, eventdata, handles)

global A
global fs
%%%%%%%%^
global timeMin
timeMin = timeSec/60;

%%%%%%%%^
global Tmin
Tmin = fs*60*5;    %% Tmin = sample data dalam 5 menit

global P_Tmin
P_Tmin = length(A);
%%%%%%%%^
axes(handles.axes4);
hold off
plot(timeMin,A);
hold on
xlabel('Menit'), ylabel('Tegangan (mV)')
title('data ECG')
grid on
%%% =====
%%% =====

```




```

%%% =====
%%% =====

%%%%>>>>> LANGKAH KE 6
%%%%>>>>> PLOTTING DATA RR INTERVAL
% --- Executes on button press in RRInterval.
function rrinterval_Callback(hObject, eventdata, handles)
% hObject      handle to RRInterval (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
global A
global fs

global thres
% thres=str2double(get(handles.TH,'string'));
%%%%^-----
global SECOND
% SECOND=str2double(get(handles.detik,'string'));

DETIK = fs*SECOND;
%%%%^-----
global AA
%%%%^-----
% global timeSec
% timeSec = (0:numel(A)-1)/fs;
%
% global timeMin
% timeMin = timeSec/60
%
% global timeOnlySomeSec
% timeOnlySomeSec = (0:numel(AA)-1)/fs;

axes(handles.axes5);
%%%%^-----
%%%%^-----
[pks,loc] = findpeaks(AA,'MinPeakHeight',thres);
%%%%^-----
%%%%^-----
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %% variabel loc DIAMBIL DARI RAW DATA ECG, MAKANYA AWALNYA 266. ARTINYA
% ITU
% %% ADALAH SAMPLE DATA KE 266 dan seterusnya
peak_start = loc(1:end-1);
peak_end = loc(2:end);
peak_to_peak = peak_end-peak_start;

global RR_int
RR_int=peak_to_peak/fs;      %%% RR INTERVAL IN SEC
R_peaks=length(loc);        %%% JUMLAH PUNCAK R YANG DITEMUKAN
%%%%-----

%%%%>>>>>>> RUMUS MEAN RR INTERVAL
mean_RR_int=mean(RR_int);   %%% MEAN RR INTERVAL
%%%%-----

```



```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% >>>>>> RUMUS BPM !!!
%% 60 detik(1 menit) DIBAGI rata-rata seluruh RR interval = BPM
%% BeatsPerMinute = round(60/mean_RR_int);
%% VAR SECOND ADALAH DURASI (dlm detik) yang ingin di analisis.
%% dibagi 60 biar jadi menit. misal, detik 300 kalo di bagi 60, jadinya 5
menit.
%% pengali itu bergantung berapa durasi data ecg (menit) yang ingin
diketahui nilainya.
pengali = SECOND/60; %%%biar jadi menit
BeatsPerMinute = round( (60*pengali)/(pengali*mean_RR_int) )

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% >>>>>> RUMUS STANDAR DEVIASI
%% 17 JUNI 2020 JAM 6 PAGI GANTI RUMUS SEK, BUAT VALIFDASI KUBIOS
%% n - 1 di GANTI JADI n DOANG
%% udah gw ganti lagi ja, 6 lewat 2 menit

n = length(RR_int);
StandarDeviasi = sqrt( sum((RR_int - mean_RR_int).^2)/(n-1) ); %%% standar
deviasi

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% rasio antara nilai mean RR dan Standar Deviasi RR
CVRR = StandarDeviasi/mean_RR_int;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
set(handles.bpm2, 'string', BeatsPerMinute)
set(handles.meanRR2, 'string', mean_RR_int)
set(handles.sdr2, 'string', StandarDeviasi)
set(handles.cvrr2, 'string', CVRR)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
hold off
plot(RR_int)%%%'MarkerFaceColor','r')
hold on
title('RR Interval')
hold on
xlabel('Sample'); ylabel('R-R(detik)')
grid on

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%>>> PROGRAM FREQUENCY METHOD
rrseries = RR_int.';
interval = 1;

Interpolasi_Badar(rrseries,interval); %> MANGGIL FUNSI INTERPOLASI
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
global hasil_int
hasil_int = ans;
global panjang_hasil_int
panjang_hasil_int = length(hasil_int); %> PANJANG DATA VARIABEL x

```

```

global absolutfft
absolutfft = abs(fft(hasil_int))/panjang_hasil_int;
%> BISA DI BAGI N, BISA JUGA TIDAK

global setengah_Y
setengah_Y = absolutfft(1:panjang_hasil_int/2+1);
%> VARIABEL AKAN MENGHITUNG DARI DATA PERTAMA absolutfft
%> cuma setengah dari keseluruhan data.

global freq
freq = linspace(0,1,length(absolutfft)); %> VARIABEL INI MENGHITUNG DARI 0
SAMPAI 1 Hz, sampai panjang Y
freq = freq(1:panjang_hasil_int/2+1); %> HANYA MENGHITUNG SETENGAH DARI
KESELURUHAN NILAI VARIABEL
% % figure('name','RR Interval')
% % plot(rrseries)
% % title('RR Interval')
% % xlabel('Sample'); ylabel('R-R(ms)')
% % grid on
% % % % .....
% % % % .....
% % PEMBAGIAN 3 JENIS FREKUENSI
% % VERY LOW FREQUENCY
global iVLF
iVLF = freq >= 0 & freq <= 0.04;
global VLF
VLF = setengah_Y(iVLF)
% % LOW FREQUENCY
global iLF
iLF = freq > 0.04 & freq <= 0.15;
global LF
LF = setengah_Y(iLF)
% % HIGH FREQUENCY
global iHF
iHF = freq > 0.15 & freq <= 0.4;
global HF
HF = setengah_Y(iHF)

axes(handles.axes6);
% % % % .....
% % figure ('name','PSD')
% % plot(freq,setengah_Y)
% % grid on
% % title('Power Spectral Density')
% % xlabel('Frequency(Hz)'); ylabel('S^2/Hz')
% % legend('PSD','VLF','LF','HF')
% % % % .....
% % % % >>> PLOTTING VLF, LF, HF
% % figure ('name','PSD')
hold off
plot(freq,setengah_Y)
hold on

plot(freq(iVLF),VLF,'-k')
plot(freq(iLF),LF,'-r')
plot(freq(iHF),HF,'-g')
hold on
grid on
title('Power Spectral Density')
xlabel('Frekuensi(Hz)'); ylabel('sec^2/Hz')
legend('PSD','VLF','LF','HF')

```



```

axes(handles.axes6);
%%%%
%%%%
%%%%
%%>> POWER DI TULIS DALAM DETIK.
%%>> KARENA RR NYA DETIK. DISERAGAMKAN.
%%>> KARENA KUBIOS SATUAN NYA MILLI DETIK. MAKA DALAM LAPORAN
%%>> SATUAN POWER DI RUBAH KE DALAM MILLI DETIK DAHULU, DENGAN CARA HASIL GUI
%%>> DIKALI 1000
global M_VLF
M_VLF = ( sum(VLF) );

global M_LF
M_LF = (sum(LF) );

global M_HF
M_HF = (sum(HF) );

set(handles.VLF2, 'string',M_VLF)
set(handles.LF2, 'string',M_LF)
set(handles.HF2, 'string',M_HF)
%%% =====
%%% =====

%%%%%%%% >>>> TOMBOL RESET GRAFIK
% --- Executes on button press in reset.
function reset_Callback(hObject, eventdata, handles)
cla (handles.axes4, 'reset');
cla (handles.axes5, 'reset');
cla (handles.axes6, 'reset');
clearStr = 'clear all';
% hObject      handle to reset (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
%%% =====
%%% =====

%%%%%%%% >>>> TOMBOL EXPORT RR INTERVAL TO TEXT FILE
% --- Executes on button press in totextfile.
function totextfile_Callback(hObject, eventdata, handles)
global A
global fs

global thres
% thres=str2double(get(handles.TH, 'string'));
%%%%%%%%^
global SECOND
% SECOND=str2double(get(handles.detik, 'string'));

DETIK = fs*SECOND;
%%%%%%%%^
global AA
%%%%%%%%^
% global timeSec
% timeSec = (0:numel(A)-1)/fs;
%
% global timeMin

```

```

% timeMin = timeSec/60
%
% global timeOnlySomeSec
% timeOnlySomeSec = (0:numel(AA)-1)/fs;

axes(handles.axes5);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[pks,loc] = findpeaks(AA,'MinPeakHeight',thres);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
peak_start = loc(1:end-1);
peak_end = loc(2:end);
peak_to_peak = peak_end-peak_start;

RR_int=peak_to_peak/fs;      %%% RR INTERVAL IN DOMAIN TIME
R_peaks=length(loc);        %%% JUMLAH PUNCAK R YANG DITEMUKAN
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%>>>>>> RUMUS MEAN RR INTERVAL
mean_RR_int=mean(RR_int);   %%% MEAN RR INTERVAL
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%>>>>>> RUMUS BPM !!!
% 60 detik(1 menit) DIBAGI rata-rata seluruh RR interval = BPM
% % BeatsPerMinute = round(60/mean_RR_int);
% % VAR SECOND ADALAH DURASI (dlm detik) yang ingin di analisis.
% % dibagi 60 biar jadi menit. misal, detik 300 kalo di bagi 60, jadinya 5
menit.
% % pengali itu bergantung berapa durasi data ecg (menit) yang ingin
diketahui nilainya.
pengali = SECOND/60; %%biar jadi menit
BeatsPerMinute = round( (60*pengali)/(pengali*mean_RR_int) )

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%>>>>>> RUMUS STANDAR DEVIASI

n = length(RR_int);
StandarDeviasi = sqrt( sum((RR_int - mean_RR_int).^2)/(n-1) ); %%% standar
deviasi

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% rasio antara nilai mean RR dan Standar Deviasi RR
CVRR = StandarDeviasi/mean_RR_int;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
set(handles.bpm2,'string',BeatsPerMinute)
set(handles.meanRR2,'string',mean_RR_int)
set(handles.sdr2,'string',StandarDeviasi)
set(handles.cvrr2,'string',CVRR)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
hold off

```

```

plot(RR_int)%%% 'MarkerFaceColor','r')
hold on
title('RR Interval')
xlabel('Sample'); ylabel('R-R(detik)')
grid on

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% EXPORT TO TEXT FILE
BVC = RR_int';
dlmwrite('RR_Interval_From_GUI.txt',BVC,'delimiter','\t','precision',3);
%%% =====
%%% =====

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%% >>>> TOMBOL UNTUK ZOOM IN GRAFIK METODE FREKUENSI
% % --- Executes on button press in ZOOM.
function ZOOM_Callback(hObject, eventdata, handles)

global A
global fs

global thres
% thres=str2double(get(handles.TH,'string'));
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
global SECOND
% SECOND=str2double(get(handles.detik,'string'));

DETIK = fs*SECOND;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
global AA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% global timeSec
% timeSec = (0:numel(A)-1)/fs;
%
% global timeMin
% timeMin = timeSec/60
%
% global timeOnlySomeSec
% timeOnlySomeSec = (0:numel(AA)-1)/fs;

axes(handles.axes5);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[pks,loc] = findpeaks(AA,'MinPeakHeight',thres);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
peak_start = loc(1:end-1);
peak_end = loc(2:end);
peak_to_peak = peak_end-peak_start;

global RR_int
RR_int=peak_to_peak/fs;      %%% RR INTERVAL IN DOMAIN TIME

```



```

R_peaks=length(loc);          %%% JUMLAH PUNCAK R YANG DITEMUKAN
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%>>>>>> RUMUS MEAN RR INTERVAL
mean_RR_int=mean(RR_int);    %%% MEAN RR INTERVAL
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%>>>>>> RUMUS BPM !!!
%% 60 detik(1 menit) DIBAGI rata-rata seluruh RR interval = BPM
%% BeatsPerMinute = round(60/mean_RR_int);
%% VAR SECOND ADALAH DURASI (dml detik) yang ingin di analisis.
%% dibagi 60 biar jadi menit. misal, detik 300 kalo di bagi 60, jadinya 5
menit.
%% pengali itu bergantung berapa durasi data ecg (menit) yang ingin
diketahui nilainya.
pengali = SECOND/60; %%%biar jadi menit
BeatsPerMinute = round( (60*pengali)/(pengali*mean_RR_int) )

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%>>>>>> RUMUS STANDAR DEVIASI

n = length(RR_int);
StandarDeviasi = sqrt( sum((RR_int - mean_RR_int).^2)/(n-1) ); %%% standar
deviasi

%% rasio antara nilai mean RR dan Standar Deviasi RR
CVRR = StandarDeviasi/mean_RR_int;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
set(handles.bpm2, 'string', BeatsPerMinute)
set(handles.meanRR2, 'string', mean_RR_int)
set(handles.sdr2, 'string', StandarDeviasi)
set(handles.cvrr2, 'string', CVRR)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
hold off
plot(RR_int)%%%'MarkerFaceColor','r')
hold on
title('RR Interval')
hold on
xlabel('Sample'); ylabel('R-R(detik)')
grid on

%%%.
%%%.
%%%.
%%% PROGRAM FREQUENCY METHOD

rrseries = RR_int.';

```

```

interval = 1;

Interpolasi_Badar(rrseries,interval); %> MANGGIL FUNSI INTERPOLASI
%%%.
global hasil_int
hasil_int = ans;
global panjang_hasil_int
panjang_hasil_int = length(hasil_int); %> PANJANG DATA VARIABEL x

global absolutfft
absolutfft = abs(fft(hasil_int))/panjang_hasil_int;
%> BISA DI BAGI N, BISA JUGA TIDAK

global setengah_Y
setengah_Y = absolutfft(1:panjang_hasil_int/2+1);
%> VARIABEL AKAN MENGHITUNG DARI DATA PERTAMA absolutfft
%> cuma setengah dari keseluruhan data.

global freq
freq = linspace(0,1,length(absolutfft)); %> VARIABEL INI MENGHITUNG DARI 0
SAMPAI 1 Hz, sampai panjang Y
freq = freq(1:panjang_hasil_int/2+1); %> HANYA MENGHITUNG SETENGAH DARI
KESELURUHAN NILAI VARIABEL

% figure('name','RR Interval')
% plot(rrseries)
% title('RR Interval')
% xlabel('Sample'); ylabel('R-R(ms)')
% grid on
%%%.
%%%.

%% PEMBAGIAN 3 JENIS FREKUENSI
%% VERY LOW FREQUENCY
global iVLF
iVLF = freq >= 0 & freq <= 0.04;
global VLF
VLF = setengah_Y(iVLF);
%% LOW FREQUENCY
global iLF
iLF = freq > 0.04 & freq <= 0.15;
global LF
LF = setengah_Y(iLF);
%% HIGH FREQUENCY
global iHF
iHF = freq > 0.15 & freq <= 0.4;
global HF
HF = setengah_Y(iHF);

axes(handles.axes6);
%%%.
% figure ('name','PSD')
% plot(freq,setengah_Y)
% grid on
% title('Power Spectral Density')
% xlabel('Frequency(Hz)'); ylabel('S^2/Hz')
% legend('PSD','VLF','LF','HF')
% %%.

```

```

%%%%%%%% >>> PLOTTING VLF, LF, HF

global batasx
batasx = str2double(get(handles.BATAS_X, 'string'));

global batasx
batasx = str2double(get(handles.BATAS_X, 'string'));

global batasx
batasx = str2double(get(handles.BATAS_X, 'string'));

%%%%%%%% figure ('name','PSD')

hold off
plot(freq, setengah_Y)
hold on

plot(freq(iVLF), VLF, 'k')

plot(freq(iLF), LF, '-r')

plot(freq(iHF), HF, '-g')
hold on

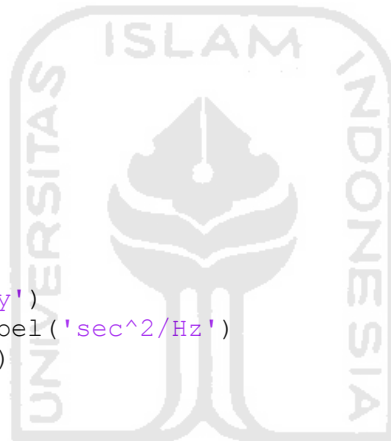
xlim([0 batasx ])
ylim([0 batasx ])

hold on
grid on

title('Power Spectral Density')
xlabel('Frekuensi(Hz)'); ylabel('sec^2/Hz')
legend('PSD', 'VLF', 'LF', 'HF')

axes(handles.axes6);
% %%%.....
% %%%.....
% figure('name','RR Interval to PSD')
% subplot(311)
% plot(rrseries)
% hold on
% title('RR Interval')
% xlabel('Sample'); ylabel('R-R(ms)')
% grid on
%
% subplot(312)
% plot(freq, setengah_Y)
% grid on
% title('Power Spectral Density')
% xlabel('Frequency(Hz)'); ylabel('S^2/Hz')
% legend('PSD', 'VLF', 'LF', 'HF')
%
% subplot(313)
% plot(freq, setengah_Y)
% hold on
% plot(freq(iVLF), VLF, '-b*', 'linewidth', 2)
% hold on
%
% plot(freq(iLF), LF, '-r*', 'linewidth', 2)
% hold on

```




```

% --- Executes during object creation, after setting all properties.
function BATAS_Y_CreateFcn(hObject, eventdata, handles)
% hObject    handle to BATAS_Y (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end
%%% =====
%%% =====

%%% ----- AKHIR FUNGSI UTAMA GUI -----
%%% -----

```

